



Atti DIDAMATICA 2022

La trasformazione digitale nella Scuola, negli ITS, nell'Università e nella
formazione professionale

Autori Vari

MILANO | NOVEMBRE 2022

Atti Convegno Nazionale DIDAMATICA 2022

36^a edizione

**Centro Congressi Fast
Milano, 10-11 novembre 2022**

**A cura di
Renato S. Marafioti, Paolo Ciancarini,
Pierfranco Ravotto e Manuel Gentile**

Chair di sessione
Viola Cadice, Paolo Ciancarini, Egidio Cipriano, Domenico Consoli, Claudio De Martini, Mara
Masseroni, Gianluca Mazzoccoli, Giorgio Mortali, Angelo Rizzo

ISBN 978-88-98091-63-8



Atti Convegno Nazionale DIDAMATiCA 2022

Centro Congressi Palazzo Fast – Piazzale Rodolfo Morandi 2, 20121 Milano

Milano 10-11 novembre 2022

A cura di: Renato S. Marafioti, Paolo Ciancarini, Pierfranco Ravotto e Manuel Gentile

ISBN: ISBN 978-88-98091-63-8

Risorse e aggiornamenti relativi a questi Atti sono disponibili all'indirizzo <https://www.aicanet.it/didamatica2022>

Copyright © 2022 AICA - Associazione Italiana per l'Informatica ed il Calcolo
Automatico Piazzale Rodolfo Morandi, 2 - 20121 Milano
Tel. +39-02-7645501 - Fax +39-02-76015717 www.aicanet.it

Licenza Creative Commons

Attribuzione - Non commerciale - Condividi allo stesso modo 3.0



Tu sei libero: di riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare quest'opera, di modificare quest'opera alle seguenti condizioni:

1) devi attribuire la paternità dell'opera citando esplicitamente la fonte e i nomi degli autori; 2) non puoi usare quest'opera per fini commerciali; 3) se alteri o trasformi quest'opera, o se la usi per crearne un'altra, puoi distribuire l'opera risultante solo con una licenza identica a questa; 4) ogni volta che usi o distribuisi quest'opera, devi farlo secondo i termini di questa licenza, che va comunicata con chiarezza.

È possibile rinunciare a qualunque delle condizioni sopra descritte se ottieni l'autorizzazione dal detentore dei diritti. Nel caso in cui l'opera o qualunque delle sue componenti siano nel pubblico dominio secondo la legge vigente, tale condizione non è in alcun modo modificata dalla licenza.

Questo è un riassunto in linguaggio accessibile a tutti del Codice Legale (la licenza integrale è reperibile su <http://www.creativecommons.it/Licenze>).

Prima edizione: novembre 2022

Editing a cura degli autori

Prefazione

A cura di Renato S. Marafioti

Milano ha ospitato la 36esima edizione della Conferenza annuale AICA. Il 10 e 11 novembre, nella splendida cornice di Centro Congressi Fast, si è discusso sulla necessità di investire sul capitale umano affinché la transizione digitale non diventi una forma di esclusione, ma si trasformi in una reale opportunità di crescita professionale. La Conferenza “La trasformazione digitale nella scuola, negli ITS, nell'Università e nella formazione professionale” organizzata da AICA - con il patrocinio del Consiglio della Regione Lombardia e di AGID, Agenzia per l'Italia Digitale ed in collaborazione con il Ministero dell'istruzione e del merito ed a FAST Federazione delle associazioni scientifiche e tecniche, nell'ambito delle celebrazioni dei suoi 125 anni – ha visto la partecipazione, tra sessioni plenarie e sessioni scientifiche parallele, di illustri esperti, docenti, ricercatori e amministratori intenti a confrontarsi e favorire una riflessione sul tema delle competenze nello scenario digitale avendo chiaro che solo attraverso una sinergia costruttiva tra mondo del lavoro, della scuola e della formazione si potranno definire le competenze necessarie in questa fase di grande innovazione.

Questa edizione si è svolta in due giorni di intensa attività e coinvolgimento, che hanno visto un proficuo confronto e un interessante scambio di idee sotto il profilo tecnico professionale, ma ancor di più personale ed umano. Credo che oggi l'Italia sconti un gap di formazione che rischia non solo di aumentare le disuguaglianze territoriali e di genere ma anche di rallentare la crescita economica del Paese. In Italia ci sono 26 milioni di persone senza competenze digitali di base, che vuol dire il 54% dei 16-74enni contro il 46% della media europea: il nostro Paese si piazza al 24° su 27 nell'indice DESI della Commissione Europea con una performance particolarmente deludente sul capitale umano. Credo che per portare avanti la trasformazione digitale nel sistema istruzione e formazione sia importante avere chiaro il contesto in cui si opera ed i suoi bisogni!”

Trasformazione digitale delle organizzazioni e della didattica, Esperienze di didattica digitale, Insegnare l'informatica, La trasformazione digitale della scuola, della formazione professionale e degli ITS – il Piano Scuola 4.0, Le Università del futuro, Coding e computational thinking, Intelligenza artificiale, realtà virtuale e realtà aumentata, Internet of things, Formazione professionale per la trasformazione digitale nella Pubblica Amministrazione, I profili professionali richiesti dal mercato, Formazione docenti e comunità di pratiche, Piattaforme

e-learning e strumenti per videoconferenze, collaborazione e condivisione sono solo alcuni degli argomenti che sono stati affrontati alla Conferenza, in cui i partecipanti, provenienti da varie aree, hanno messo in comune le loro competenze e la loro esperienza uniti nella volontà di trovare risposte chiare ed esaurienti ad una delle questioni più attuali del nostro presente.

L'edizione 2022 di Didamatica, articolata in due giornate di studi, con circa 40 relatori ed oltre 90 paper presentati, ha messo in risalto che la consapevolezza, la formazione, il continuo aggiornamento professionale e lo scambio di informazioni siano gli strumenti più efficaci per far fronte ai problemi del digital divide. Il processo di digitalizzazione della didattica e dell'organizzazione scolastica italiana ha preso avvio da oltre 15 anni con l'attuazione delle prime misure di trasformazione digitale. Oggi, con i fondi del Piano Nazionale di Ripresa e Resilienza e dei fondi strutturali europei della programmazione 2021-2027 questo processo conosce un completamento e, al tempo stesso, un nuovo, forte impulso, sia per la rilevanza degli investimenti sia per l'approccio sistemico delle azioni. Credo che le sfide che aspettano il sistema Paese nei prossimi anni siano di sicuro rilievo e riguarderanno sicuramente la gestione del cambiamento introdotto dal Covid-19 nel mondo sociale e professionale, e la trasformazione del Paese verso nuovi orizzonti in termini di digitalizzazione, transizione ecologica, mobilità sostenibile, istruzione, inclusione e coesione sociale e salute, tutti obiettivi delle linee di sviluppo e investimento individuate dall'UE per il rilancio dell'economia e della vita in Europa. Il PNRR individua nelle persone, prima ancora che nelle tecnologie, il motore del cambiamento e dell'innovazione. Il progresso tecnologico può favorire lo sviluppo umano e contribuire a creare condizioni ottimali per l'esercizio dei diritti dell'uomo. Al contempo, il suo uso ed eventuale abuso hanno implicazioni più ampie per i valori fondamentali delle società democratiche, tra cui l'uguaglianza e la correttezza.

Il comitato è un organismo, previsto dall'ordinamento giuridico, che persegue uno scopo altruistico, di solito di pubblica utilità come promuovere una determinata iniziativa culturale (Didamatica, Congresso, etc.) da parte di una pluralità di persone.

Comitato scientifico

Si tratta di un'area dei lavori, comprende persone di elevata professionalità per fornire gli orientamenti principali sui quali si svilupperà Didamatica o il Congresso. Ha il compito di stimolare, gestire e promuovere, attraverso le idee e le iniziative dei propri membri, attività per valorizzare e promuovere la cultura del "fare rete". È un organismo con un ruolo propositivo e consultivo ai fini progettuali per: garantire ed assicurare la qualità tecnico scientifica dei contenuti trattati e pubblicati, supportare l'associazione nella definizione degli argomenti da approfondire, nell'analisi di efficienza e efficacia dei temi scelti, valutare le proposte, selezionare e contattare i relatori, monitorare e validare il corretto svolgimento dei lavori e dello sviluppo del portale, riconoscendo l'impegno di tutti i partner coinvolti.

Renato S. Marafioti, Presidente AICA, Associazione Culturale Format, Chair
Paolo Ciancarini, Università Bologna
Mario Allegra, Direttore iTD-CNR
Diego Antonini, Presidente Assinter, FVG
Ivana Cipolla, AICA
Laura Biancato, DS Veneto
Michele Da Col, FVG
Alberto Felice De Toni, Università Udine
Bruno Lamborghini,
Valerio Libralato,
Franco Patini,
Susanna Sancassani, Direttrice di METID, Politecnico di Milano
Giuseppe Fiorentino, Presidente AIUM (Associazione Italiana Utenti Moodle)
Pierfranco Ravotto, AICA, Direttore Bricks
Claudio Demartini, Politecnico Torino
Mariacarla Calzarossa, Università Pavia
Viola Schiaffonati, Politecnico Milano, Direttore Mondo Digitale
Ferro Nicola, Università Padova
Gerald Futschek, CEPIS
Claudio Pardini,
Luigi Laura, Uninettuno Roma, Presidente Comitato Olimpiadi Informatica
Anna Vaccarelli, CNR Pisa, Registro.it
Stefano Paraboschi, Presidente GII
Ernesto Damiani, Presidente CINI e Università Milano
Claudia Sartirani, PMI DAY, vice presidente della Piccola Industria di Confindustria Bergamo
Amanda Ferrario - Dirigente Scolastico ITE TOSI Busto Arsizio
Silvestro Cozzolino - Vicepresidente Sezione AICA Lombardia
Mattia Monga – CINI
Enrico Nardelli, Università di Roma Tor Vergata

Comitato organizzatore

È formato da tutti i componenti di elevata professionalità che a livello centrale, AICA e MIUR, e a livello locale, Università, USR, Confindustria, Regioni, si occupa di: richiedere contributi, selezionare patrocini, redigere il programma centrato sul tema e sulle tracce selezionate, contattare i relatori, identificare e risolvere eventuali problemi logistici, presidiare il budget dei costi, sollecitare sponsorizzazioni-contributi, gestire i contenuti sul portale AICA.

Renato S. Marafioti, Presidente AICA, Associazione Culturale Format, Chair

Alberto Pieri, FAST

Pierfranco Ravotto, AICA, Direttore Bricks

Salvatore Garro, Referente con le Istituzioni, AICA

Viola Cadice, AICA Lombardia

Stefania Dimatteo, AICA - coordinamento operativo

Elisabetta Benetti, AICA - aggiornamento contenuti portale AICA

Manuel Gentile, ITD-CNR sezione di Palermo - Responsabile gestione piattaforma EasyChair

Amanda Ferrario - Dirigente Scolastico ITE TOSI Busto Arsizio

Silvestro Cozzolino - Vicepresidente Sezione AICA Lombardia

Carlo Bellettini – CINI

Comitato di programma

Si tratta di un comitato d'onore, composto da persone di elevata professionalità che determina lo scopo e il modo per perseguire le finalità delle manifestazioni. Ha il compito di creare il collegamento tra l'Associazione e i terzi; mira al conseguimento di obiettivi rilevanti dal punto di vista sociale o di interesse pubblico. Si occupa dei seguenti compiti: scelta del titolo/tema trainante del convegno, identificazione delle tracce principali sulle quali si svilupperà il programma dell'evento, preparazione della call for paper, costituzione del Comitato Scientifico e Organizzatore.

Renato S. Marafioti, Presidente AICA, Chair
Manuel Cacitti, Vicepresidente Vicario AICA, Co-Chair
Gaetano Di Bello, Vicepresidente AICA, Co-Chair
Antonio Piva, Vicepresidente AICA, Co-Chair
Giuseppe Mastronardi, Presidente Onorario AICA, Politecnico di Bari
Giovanni Adorni, Università degli Studi di Genova
Anna Brancaccio, Ministero Istruzione
Eleonora Caracciolo, AICA Calabria
Margherita Lanna, Aica Campania
Franco Paris, AICA Emilia Romagna – Marche
Giovanni Franza, AICA Internazionale
Luciano Lattanzi, AICA Lazio
Viola Cadice, AICA Lombardia
Raffaele Perrotta, AICA Nord-Est
Marina Marchisio, AICA Piemonte
Angelo Grana, AICA Puglia
Enrico Bocci, AICA Toscana
Angelo Rizzo, AICA Valdadige
Claudio Demartini, Consigliere eletto
Antonella Evangelista, Consigliere eletto
Alberto Graziani, Consigliere eletto
Emanuela Scalzotto, AICA, Consigliere eletto
Carlo Tiberti, AICA, Presidente Sezione AICA Lombardia
Paolo Schgör, Direttore AICA
Michele Petrocelli, USR Lombardia, Ufficio XII Pavia
Elena Ferrara, senatrice
Anna Vaccarelli, CNR Pisa, Registro.it
Michele Petrocelli - USR Lombardia - Ufficio XII Pavia
Amanda Ferrario - Dirigente Scolastico ITE TOSI Busto Arsizio

INDICE

CAPITOLO 1 TRASFORMAZIONE DIGITALE DELLE ORGANIZZAZIONI E DELLA DIDATTICA	13
La Coppa Student.....	14
Gestione del progetto di prototipi per la digitalizzazione della carriera scolastica, della	18
Un sistema software per la gestione a distanza dei laboratori informatici multidisciplinari	28
La Formazione Digitale del Musicista	35
Il progetto Meteo Vasari e la Didattica Digitale.....	45
Live Vasari: una risposta didattica-digitale alla.....	55
Scienze per il Futuro, un percorso STEM.....	59
Certification system based on Blockchain and Open Badge technology	67
Montagna ONU2030&PNSD, i laboratoriSTEAM aiutano alla riscoperta della montagna	73
La competenza digitale convergente: prospettive di educazione e formazione presso il Polo Formativo Scolastico di Treviglio.....	84
Trashware: utilizzo responsabile delle risorse scolastiche	94
Analisi sul ruolo centrale dei dati del sistema scolastico italiano per completare una piena transizione digitale	103
La facilitazione grafica mediata digitalmente.	112
LaTeX tra competenze digitali e accessibilità: un'esperienza di PCTO con il Laboratorio Polin.....	121
Tra Digital Soft Skill e Formazione da Remoto. Nuovi modelli didattici per nuove competenze	131
Progetto Biblioteca IC Poppi	141
Aula Phygital	145
Équipe Formativa Territoriale: come è nata e come si è trasformata in Piemonte	154
La valutazione e l'autovalutazione negli ambienti didattici integrati.....	162
Tecnologie e metodologie didattiche nell'Università del post-COVID. Un'esperienza formativa con dottorandi e neo-ricercatori.	170

CAPITOLO 2 ESPERIENZA DIDATTICA DIGITALE 174

Materiale formativo ICDL adeguato a persone non vedenti, sorde o che hanno difficoltà di comunicazione 175

Un'esperienza didattica di utilizzo delle TIC per promuovere l'apprendimento delle competenze emotive a supporto degli studenti nella scuola della post-pandemia 185

Il digitale nella didattica: da zattera indispensabile durante la pandemia, a mezzo essenziale con cui recuperare una nuova normalità 195

Artsteps. Un'esperienza di progettazione culturale e virtuale¹ 205

Volontari (digitali) per l'Educazione 213

Futuri insegnanti di sostegno. Gli Smart VisualMedia per l'inclusione scolastica degli alunni..... 218

Game – based learning in didattica: progettare escape room con gli studenti universitari 228

Media Education: un'esperienza didattica con gli adulti dei corsi serali 234

Introduction to Data Literacy with Tableau in HighSchool 239

CAPITOLO 3 INSEGNARE L'INFORMATICA..... 246

Un Percorso di Machine Learning attraverso la piattaforma Open Source KNIME 247

PythagorHub: quando Pitagora incontra Android..... 258

Empowering to code a diverse population of future digital designers..... 268

Innovative learning experiences: the “Wellbeethon” Marathon at the University of Foggia 278

Didattica ibrida dell'Informatica in un Liceo Scientifico delle Scienze Applicate: condivisione dei corsi con MoodleNet Central..... 283

Bee Environmental Monitoring 288

Laboratorio di #bioinformatica: un PCTO innovativo in periodo pandemico e post pandemico 296

Di cosa parliamo quando parliamo di “programmi” * 306

Tirocini informatici online in Istat: esperienze di learning-by-doing di coding e di utilizzo di strumenti low-code 314

Enhancing a block-based IDE to improve learning of computer programming for people with and without dyslexia and/or dyscalculia..... 322

Less is More, un approccio metodologico al microlearning nella scuola 330

CAPITOLO 4 CODING E COMPUTATIONAL THINKING	335
Il coding: strumento utile per la didattica multimediale e il team working	336
Coding e pensiero computazionale nella didattica digitale come metodo e pratica multidisciplinare.....	344
Robotica educativa e sostenibilità ambientale: la “Serra Otonga”	354
Un artefatto cognitivo per la costruzione delle social skills. Il pensiero computazionale ed il pensiero creativo nel biennio del liceo scientifico	364
Computing Stories Storie narrate in codice	374
Sviluppo del pensiero computazionale nella scuola dell’infanzia e nella scuola primaria.....	381
Struttura e resoconto del primo anno di sperimentazione curricolare del live coding con il software Sonic Pi: implicazioni riguardanti la definizione di pensiero computazionale e sulla programmazione nelle discipline artistico espressive	391
Il concetto di “errore di programmazione” nelle parole degli insegnanti: uno studio esplorativo.....	402
Spiegare il “caso” e l’array nel live coding musicale.....	410
CAPITOLO 5 INTELLIGENZA ARTIFICIALE, REALTÀ VIRTUALE E REALTÀ AUMENTATA E INTERNET OF THINGS	420
Sperimentazione didattica innovativa con utilizzo di tecnologie digitali	421
AI_chat. Chatbot per lo sviluppo dell’interazione in apprendenti principianti di italiano L2.....	424
METAV-EDU - come utilizzare il metaverso a.....	428
AI: a super-tool for fostering human thought.....	437
A scuola di Intelligenza Artificiale	445
CAPITOLO 6 FORMAZIONE DOCENTI E COMUNITÀ DI PRATICHE.....	455
Did@ttiva: una comunità di pratica.....	456
La robotica educativa nelle scuole dell’infanzia. Un’esperienza di formazione insegnanti online.....	469
A blended programme to foster interdisciplinarity in Urban forestry: the Uforest experience	479
La sfida dig-etica e le lenti di Guglielmo.	483
Dalla scuola a distanza all’ecosistema digitale: un progetto di formazione e ricerca nel Canton Ticino	491

CAPITOLO 7 PIATTAFORME ELEARNING E STRUMENTI PER VIDEOCONFERENZE, COLLABORAZIONE E 501

Training e Simulazione per Migliorare la Gestione della Sicurezza negli Ospedali 502

Come continuare a usare sul serio le prove di Moodle anche dopo l'emergenza Covid 507

The design of gender-balanced educational digital content: a MOOC to raise awareness about gender issues.
..... 519

Virtual Learning Environments for Remote Lecturing: Comparing Mozilla Hubs and Gather 523

CAPITOLO 8 GAME E DRONI 533

Un videogioco per educare allacybersecurity 534

MyLIFE – The Game: una piattaforma on-lineal servizio della scrittura autobiografica 539

“Volare con APR di ultima generazione” 545

“Il Drone amico” 554

I droni, gli adolescenti, il volo e la filosofia 564

Enhancing a block-based IDE to improve learning of computer programming for people with and without dyslexia and/or dyscalculia

Stefano Federici, Elisabetta Gola, Elisabetta Sergi and Martina Giorgi

University of Cagliari, Cagliari, Italy
sfederici@unica.it, egola@unica.it

Abstract

In this paper we describe the early stages of a project to make learning of computer programming easier for people with and without dyslexia/dyscalculia. Several additions to classic IDEs for block-based programming environments, based on general directions to improve learning for people with learning disabilities such as dyslexia/dyscalculia, are proposed. The proposed additions will be evaluated in a large test on students with and without dyslexia/dyscalculia from different school levels and grades, based on the general finding that, when reviewing specific topics with LD students, the style of explanation they required always brings improvements to lessons and course resources.

1 Introduction



Figure 1: The ScratchJr programming language



Figure 2: The Scratch programming language



Figure 3: The App Inventor programming language

Teaching of computer programming has greatly improved in the last two decades thanks to new simplified but -more and more- powerful environments such as ScratchJr (<http://scratchjr.org/>, 2014, Figure 1), Scratch (<http://scratch.mit.edu>, 2007, Figure 2) or App Inventor (<http://appinventor.mit.edu>, 2010, Figure 3).

In these new visual environments, users are supported in their learning process by having everything they need already available and clearly organized in a simplified view so that -for the most part of the time- they have just to drag and snap together visual elements and colored blocks. These visual features, that are available to all kind of users, and are clearly useful to everyone (Resnick et al., 2009), are in the direction of what are the visual suggestions to improve learning for people with LDs (Learning Disabilities). But most of the difficulties of dyslexic and dyscalculic people, such as problems with reading, writing, and remembering words and numbers, and poor short-term memory (Powell et al., 2003; Ghani and Gathercole, 2013; Stienen-Duranda and George, 2014), are not coped well with by programming languages (Fuertes et al., 2016; Gonzàles, 2017).

Solving these problems for people with LDs will end up -it is our hope- in better introductory tools for computer programming even for people without LDs. Indeed, as clearly stated by Emmanuel Schanzer, founder and co-director of Bootstrap -a research project based out of Brown University that develops computer-science curricular modules to be used within K-12 classes in math, physics, and other subjects- “the guiding principle should be “everyone takes it, and everyone takes it together”” so that “whenever you end up solving an accessibility challenge, you end up solving other problems you didn’t even know you had” (Herold, 2018).

This has also been a general finding in the course of Foundations of Computer Science of the degree of Communication Studies of the University of Cagliari. When students with LDs have asked to review specific topics, the style of explanation they required has always been found generally clearer than the previous one, and when ported to the course lessons and resources there has always been a general improvement.

In the following sections we will describe what features of standard or enhanced block languages are already available to support people with dyslexia/dyscalculia, and what instead must still be added by proposing several further enhancements to the most popular block languages. We will illustrate how the usefulness of these additions are going to be tested in the project, both in elementary and middle school students with dyslexia. Finally, we will describe how the usefulness of these additions is going to be tested even for students without dyslexia.

2 Standard block languages' features and dyslexia

Scratch, but also block-languages in general and their extensions, already have a lot of features that can cope with common problems of people with LDs such as dyslexia. Usually, people with dyslexia have several strengths compared to their peers with no LDs (Everatt et al., 2007) such as a better visual-spatial memory and higher creativity, but they show a number of weaknesses that have an impact on those skills that are classically related to computer programming, such as difficulties in reading and writing the commands of the programming language by using a keyboard, in remembering the choices made previously in order to coherently use them in the whole project, in coping with the large amount of features of a programming environment. Some, or part of, those difficulties are coped with by visual programming languages based on the metaphor of building blocks. We will start by seeing in detail what are the features offered by programming environments such as ScratchJr, Scratch and App Inventor and what problems of people with dyslexia they can already solve.

2.1 Font size

Even if block-based languages are visual programming languages, nevertheless the meaning of each programming block is still mostly described by written words. Being reading text one of the main problems of dyslexic people (Lyon et al., 2003) every improvement that can be made on the font used by their programming tool would be very important (Rello and Baesa-Yates, 2013). Being the new block-based programming tool based on browsers, even if it is not immediate to change the font of a web page, luckily it is very easy to change at least the size of the font by adjusting the page zoom. Adjusting the zoom of the page is not ideal, as it changes a lot more than just the font size, but even a slight change of the font size can be critical for dyslexic people and help them in making the text more readable (O'Brien et al., 2005).

2.2 Sprites, Components and Blocks

The basic components of the outcome of programs created by using those programming environments are animated images (Scratch's and ScratchJr's *sprites*) or clickable elements on a smartphone screen (App Inventor's *components*). Moreover, the commands of block languages are *blocks* that can be dragged and snapped together to create programs. Users can then test the effect of each command by clicking the block and looking at what happens in the IDE. Even if there is still non-conclusive evidence on this field (Marshall, 2007; Falcão and Price, 2010; Jamali et al., 2018) the usage of interfaces with tangible elements has historically been seen as an important feature for dyslexic users.

2.3 Colored blocks

In block-based programming environments, blocks (i.e., commands) are grouped in categories, each

category collecting commands that act on different aspects of the sprites or components (i.e., movement, looks, trigger actions, etc. in Scratch and ScratchJr; control, logic, math, text, etc. in App Inventor). Each category has its own color, clearly differentiated from other categories' colors. The usage of different highlight colors for dyslexic people has been shown to be a relevant feature to make reading easier (Pinna and Deiana, 2018).

2.4 Visual blocks

Reading and writing words is the main difficulty for dyslexic people. Every command in a block-based language is a ready-to-use block, that does not require the user to remember how the command is written. The visual nature of blocks makes computer languages' "syntax" completely irrelevant.

2.5 Spatial organization

Programs in block languages are composed by many small units, called *scripts*. Scripts can be freely moved in the coding area so that many scripts, if necessary, can be arranged side-by-side. This is completely different from classic IDEs for text-based languages, where lines of code are usually arranged vertically. This feature of block-based programming languages can build upon the ability of dyslexic people to remember and visualize large, interconnected pictures of whole problems (Everatt et al., 2007; Stein, 2018).

2.6 Self-populating menus

When new elements are added to a block language, such as a new Sprite, a new Costume, a new Sound, or a new Variable, the name of the element is automatically added to the block menus where that element can be selected without having to retype its name. This makes easier for dyslexic people to cope with possible problems of short-term memory (Hammond, J., and Hercules, 2003), as they don't have to remember exactly how they have called something (Powell et al., 2003).

3 Dyslexia and enhanced block languages features

Whereas, as we have seen, many difficulties of LD users are coped with by modern block-languages, a lot of other difficulties are left unheard. Block environments' IDEs are not fully ready for dyslexic people as they do not take into account their difficulty at reading standard fonts with a standard spacing, at using the standard black/white background colors, at handling cognitive load without being overwhelmed, etc. Some, or part of, those difficulties are coped with by several enhancements of block languages, such as the ScratchAddons extension (<https://scratchaddons.com/>), the Snap block-based programming language (<https://snap.berkeley.edu/>), and the ScratchJr block-based programming language.

ScratchAddons is a browser extension that allow to selectively enable UI addons for the online version of the Scratch 3 editor. Snap instead is an enhanced version of Scratch, with advanced programming features. Despite this, Snap's interface is very similar to Scratch's and the full power of the tool is mostly hidden under the hood. Finally, being ScratchJr an alternative version of Scratch oriented to young people that can easily understand images but do not understand written words yet, it could be of help to dyslexic people as they can be helped by images when thinking (Hammond and Hercules, 2003).

Let us see in detail what are the features offered by the many addons of ScratchAddons, the extra features of Snap, and the picture-based philosophy of ScratchJr that are relevant to people with dyslexia.

3.1 Text highlight

The "Customizable block text style" addon of ScratchAddons allow the user to change the thickness and, optionally, to add a shadow to the text of the blocks. This feature can hinge on the fact that a boldface font can improve comprehension for dyslexic people (Rello et al., 2014).



Figure 4: Comparing the standard Scratch text (left) and the thicker and shadowed text (right) enabled in ScratchAddons

3.2 Custom spacing in blocks

The “Customizable block shape” addon allow the user to change the spacing around the block text, the size of the corners, the size of the “hooks” that allow to precisely align blocks also making available several pre-settings for standard shapes (e.g., Scratch 2.0 thinner blocks, flat blocks, etc.). This feature can solve the problem that several dyslexic people have with line spacing (Madhavan et al., 2016).



Figure 5: Comparing the standard Scratch block shape (left) and a thinner shape (right) created in ScratchAddons

3.3 Hiding UI areas

Even if the Scratch UI is kept, on purpose, as clean and simple as possible, even non-dyslexic people feel that it is sometime very crowded, especially when they are using smaller devices such as tablets. This is clear to Scratch developers too, that indeed, since the initial version of Scratch, allowed the users to reduce the size of the Stage. But, even if useful, this feature is not enough. For this reason, the “Auto-hiding block palette” and “Hide stage and sprite pane” addons of ScratchAddons allow the user to hide automatically (or on request) the block palette at the left-hand side of the tool and the Stage and the sprite pane at the right-hand. By doing this, the user has a bigger central area where scripts can be composed and looked at in an easier way. In the experiment, we will investigate if this addons allow to reduce the sense of visual crowding (Bellocchi, 2013) that some dyslexic people feel when the UI has less space than desired.



Figure 6: Comparing the standard Scratch UI (left) and the larger script area (right) created in ScratchAddons

3.4 Text colors

The “Customizable block colors” addon allow the user to select their favorite colors for each block category and to set to black the color of the text of the blocks instead of the white standard color. This allow to fully customize the background colors of the block text as the ideal color combination

for several dyslexic is black on a cream background (Rello and Bigham, 2017). In the experiment we will investigate if there is a beneficial effect of alternative color combinations with respect to the standard colors, trying to get an effect closer to the effect of “warm” colors found by Rello and Bigham (2017).



Figure 7: Comparing the standard block colors (left) and the alternative colors (right) created in ScratchAddons

3.5 Amount of available blocks

Even if Snap managed to keep to a minimum the number of additional blocks that allow the users to create program by using its really advanced mechanisms, Snap users have started to ask for a way to hide unnecessary blocks from the block’s palette. This feature, that is handy for teachers of young users to avoid that they are confused by blocks that they never have to use, is something that can certainly help dyslexic users too to reduce their cognitive load. We will be investigating this point too in the experiment.

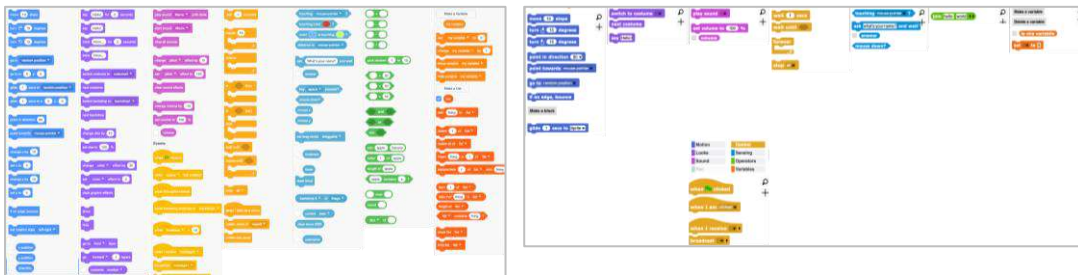


Figure 8: Comparing the standard set of Scratch blocks (left) to a reduced set of blocks allowed by Snap (right)

3.4 Blocks represented by images

ScratchJr, being a specific evolution of Scratch designed for young children that cannot read yet, uses blocks whose behavior is represented by a single image/symbol instead of words (ScratchJr, 2022). As dyslexic people are usually helped by using mental maps containing links and drawings, we have made the hypothesis that an image-only -or text-plus-image- representation of the programming blocks could make easier to understand and remember the meaning of the blocks.

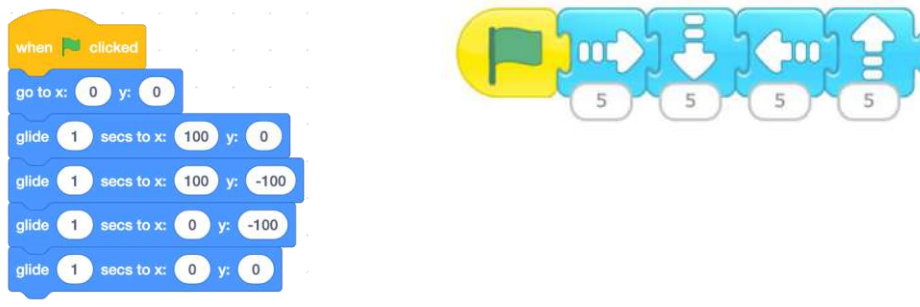


Figure 9: Comparing the a “textual” Scratch’s script (left) to a symbolic ScratchJr’s script (right)

4 A new proposal for enhanced block languages features

Even if many difficulties of LD users are coped with by the standard or additional features of enhanced block-languages, something is still missing. For example:

- even if studies show that some of the common fonts allow for best reading performance of dyslexic people (Rello and Baesa-Yates, 2013), some Scratch users think that using new, specifically designed fonts for dyslexic, could improve their usage of the tool (Scratch Forum, 2020)
- several studies show that listening comprehension of people with dyslexia is very often not below the average (Hirschmann et al., 2021) and that dyslexic students using audiobooks can significantly improve their reading accuracy (Everatt et al., 2007)
- whereas correctly writing a number is a complex task for people with dyscalculia, repeating a number out loud is, comparatively, fairly easy to them.

Hinging on these studies and hypotheses, we thought that it was worth to investigate several further improvements that can be added to a block language.

4.1 Font

As already said, block-based programming tools -as of today- do not allow their users to change the font of the tool. Furthermore, standard browsers do not allow to change the font of the webpage. However, it is possible:

- to install fonts specifically studied for dyslexic people, such as the OpenDyslexic font (<https://www.dafont.com/open-dyslexic.font>)
- to install browser extensions that allow to change the browser font, such as the Font Changer extension available in the Chrome Web Store

Thanks to these possibilities we will be investigating if a font specifically designed for dyslexic people can make it easier using a block-language programming tool.



Figure 10: Comparing the standard font of Scratch (left) to the OpenDyslexic font (right)

4.2 Speaking blocks

Scratch makes available to its users an extension that allow programs to talk. Unfortunately, this is not true for regular blocks and scripts. A “talking” script, that is a script whose blocks will be read by a synthesized voice when asked by the user, either by selecting a specific option in the contextual menu, or by pressing a specific click+key combination, could make the script more easily comprehensible to dyslexic users.

4.3 Speech recognition

Allowing all numeric and textual values, used in Scratch as arguments of its blocks, or as settings in the sprite info area, to be inputted by using the user voice, something that we do today very frequently by using our smartphones, could make creating correct scripts easier for dyscalculic and dyslexic users. The previous “speaking blocks” feature could then allow them to check that the value of the arguments they have entered by using their voice are exactly the expected ones.

5 The experiment

The experiment will be conducted on LD (dyslexic/dyscalculic) and non-LD students of 10 elementary and 10 middle schools of the Cagliari school district, each school having an average of 10-20 classes. Students will be thought a minimal set of Scratch and ScratchJr blocks and their behavior. Then LD and not-LD students will be tested for the speed and the precision they will show in understanding simple Scratch/ScratchJr scripts and the corresponding simple algorithms written in a pseudo-language very close to natural language. Each test will be done in different setups, where tool fonts, visible areas, and advanced features will be enabled or disabled as necessary. Whenever possible, the scripts and algorithms will be shown in real tools, such as:

- Scratch
- Scratch with specific addons enabled
- ScratchJr
- Snap

In cases where the modification of the tools would be too complex before knowing if the proposed features are of real help to LD users, the new features and additions will be tested by using mockups or images. The questions to which the test will answer are:

1. do the features already available in block-languages make computer programming easier to people with dyslexia?
2. do the features already available in *enhanced* block-languages make computer programming even easier to people with dyslexia?
3. do the proposed features, not yet available in block-languages, make computer programming even easier to people with dyslexia/dyscalculia?

Finally, students will be asked which ones of the several settings seem better to them, in order to evaluate which settings are putting them more at ease. The test will be then repeated on non-dyslexic/dyscalculic students to evaluate the eventual improvements for them too.

6 Conclusions

Evaluating on non-dyslexic/dyscalculic people the better settings for a block-language tool is the final goal of this study. The hypothesis, that we hope to be confirmed, is that most -if not every-addition specifically studied for dyslexic/dyscalculic users makes the tool easier for all students, regardless of they having LDs or not.

References

- Bellocchi, S. (2013). Developmental Dyslexia, Visual Crowding and Eye Movements. In *Eye Movement: Developmental Perspectives, Dysfunctions and Disorders in Humans*. Nova Science Publishers
- Everatt, J., Weeks, S., and Brook, P. (2007). Profiles of Strengths and Weaknesses in Dyslexia and Other Learning Difficulties. In *Dyslexia, An International Journal of Research and Practice* (pp. 16-41). Volume 14, Issue 1. <https://doi.org/10.1002/dys.342>

Falcão, T.P., Price, S. (2010). Informing design for tangible interaction: a case for children with learning difficulties. In *Proceedings of the International Conference on Interaction Design and Children*. ACM.

Hammond, J. and Hercules, F. (2003). *Understanding Dyslexia*. The Glasgow School of Art. Retrieved from <https://www.ed.ac.uk/files/imports/fileManager/Understanding%20Dyslexia%20Inc%20Addendum%20Feb%202015.pdf>

Herold, B. (2018, October). How to Design Computer Science for Students with Disabilities. Retrieved from <https://www.edweek.org/teaching-learning/how-to-design-computer-science-for-students-with-disabilities/2018/10>

Hirschmann, M., Farris, E. A., Alexander, E., Flipse, J., and Odegard, T. (2021). *SCREENING FOR RISK AND CHARACTERISTICS OF DYSLEXIA*. Middle Tennessee State University. Retrieved from <https://www.mtsu.edu/dyslexia/ScreeningforDyslexiaBook.pdf>.

Jamali, S. N., Admodisastro N., Hamid, S. S. A., Kamaruddin, A., Ghani, A.A.A., and Hassa, S. (2018). Towards Designing Tangible Interaction for Children with Dyslexia in Learning the Malay Language. In *Recent Advances on Soft Computing and Data Mining*. Springer.

Lyon, G. R., Shaywitz, S. E., Shaywitz, B. A. (2003). A definition of dyslexia. In *Annals of Dyslexia*, Volume 53, Issue 1, pp. 1-14

Madhavan, I., Sharanjeet-Kaur, Hairol, M. I., Mohammed, Z. (2016). Spacing improves reading in dyslexic children. In *Asia Pacific Journal of Developmental Differences* (pp. 3-20). Volume 3. Issue 1. DOI: 10.3850/S2345734114000183

Marshall, P. (2007). Do tangible interfaces enhance learning? In *Proceedings of the International Conference on Tangible and Embedded Interaction*. ACM.

Milani, A., Lorusso, M. L., Molteni, M. (2010). The Effects of Audiobooks on the Psychosocial Adjustment of Pre-adolescents and Adolescents with Dyslexia. In *Dyslexia* , Volume 16, Issue 1. pp. 98-97.

O'Brien, B. A., Mansfield, J. S., and Legge, G. E. (2005). The effect of print size on reading speed in dyslexia. *Journal of Research in Reading*, Volume 28, Issue 3, pp. 332–349

Pinna, B., and Deiana, K. (2018). On the Role of Color in Reading and Comprehension Tasks in Dyslexic Children and Adults. In *i-Perception* (pp. 1-22). Volume 9, Issue 3.

Powell, N., Moore, D., Gray, J., Finlay, J., Reaney, J. (2003). *DYSLEXIA AND LEARNING COMPUTER PROGRAMMING*. In *Proceedings of the 4th Annual Conference of the Learning and Teaching Support Network Centre for Information and Computer Sciences*. Galway. <https://doi.org/10.11120/ital.2004.03020005>

Rello, L., and Bigham, P. J. (2017). Good Background Colors for Readers: A Study of People with and without Dyslexia. In *Proceedings of The 19th International ACM SIGACCESS Conference on Computers and Accessibility* (pp. 72-80). ACM. <https://doi.org/10.1145/3132525.3132546>

Rello, L., Saggion, H., and Baeza-Yates, R. (2014). Keyword Highlighting Improves Comprehension for People with Dyslexia. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations* (pp. 30–37), Association for Computational Linguistics.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., and Kafai, Y. (2009) Scratch: programming for all. In *Communications of the ACM*, Volume 52, Issue 11, pp. 60–67. <https://doi.org/10.1145/1592761.1592779>

Scratch Forum (2020). Dyslexic Font. Retrieved from <https://scratch.mit.edu/discuss/topic/411561/>

ScratchJr (2022). Blocks descriptions. Retrieved from <https://www.scratchjr.org/learn/blocks>

Stein, J. (2018). Why dyslexics make good coders. Retrieved from <https://www.bcs.org/articles-opinion-and-research/why-dyslexics-make-good-coders/>.