

## Resources for Getting Kids into Computing

### Resources for Getting Kids into Computing

2024 NCHEA Conference and Curriculum Fair

Tyler Morten

#### Who Am I?

- Tyler Morten
- Believer in Jesus Christ
- Husband and father of 6 kids (3 currently homeschooled)
- Part of my ministry is helping with IT here at IHCC
- Contact me
  - @tylertorten on X

#### Who Is This Workshop For?

- You have homeschooled kids.
- They have an interest in how things work or creative mind.

#### Purpose of the Workshop

- Help finding resources to encourage your kids further into computers.
- Teach you how to spot the best resources out there.

#### Agenda

1. Getting Started
2. Keeping it Going
3. Recommended Resources

### Getting Started

#### Priorities

- Faith in Christ
  - Mark 8:36 "For what does it profit a man to gain the whole world, and forfeit his soul?"
- Learning God's Word
- Our discoveries about God's creation cause us to glorify Him.

#### How I Got Started

My First Computer



Around age 12

IBM PS/1



### Q-Basic

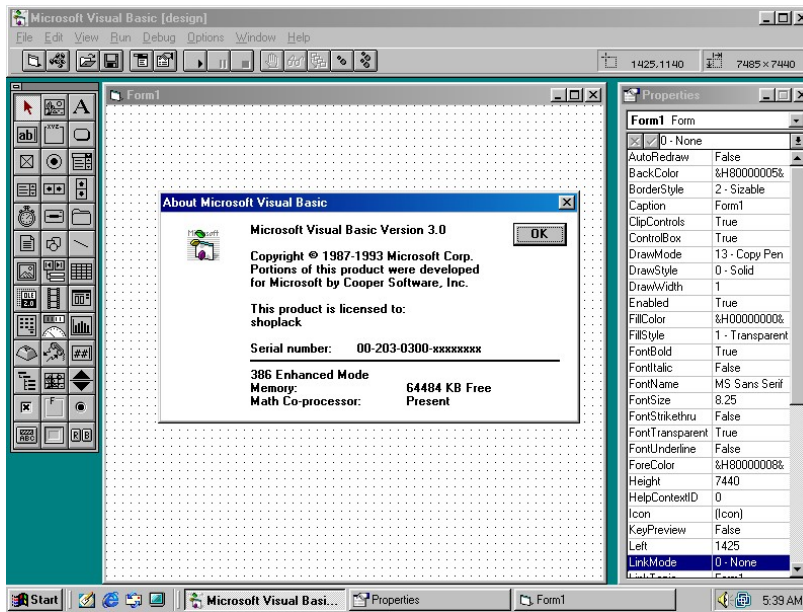
Making Games

```
File Edit View Search Run Debug Options Help
MIBBLES.BAS
-----
Q B a s i c  N i b b l e s
Copyright (C) Microsoft Corporation 1990
Nibbles is a game for one or two players.  Navigate your snakes
around the game board trying to eat up numbers while avoiding
running into walls or other snakes.  The more numbers you eat up,
the more points you gain and the longer your snake becomes.

To run this game, press Shift+F5.
To exit QBasic, press Alt, F, X.
To get help on a BASIC keyword, move the cursor to the keyword and press
F1 or click the right mouse button.
-----
Immediate
<Shift+F1=Help> <F6=Window> <F2=Subs> <F5=Run> <F8=Step> 00001:001
```

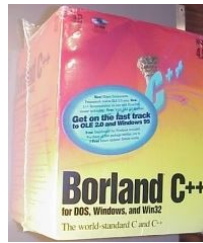
### Visual Basic (GUI) Age 13

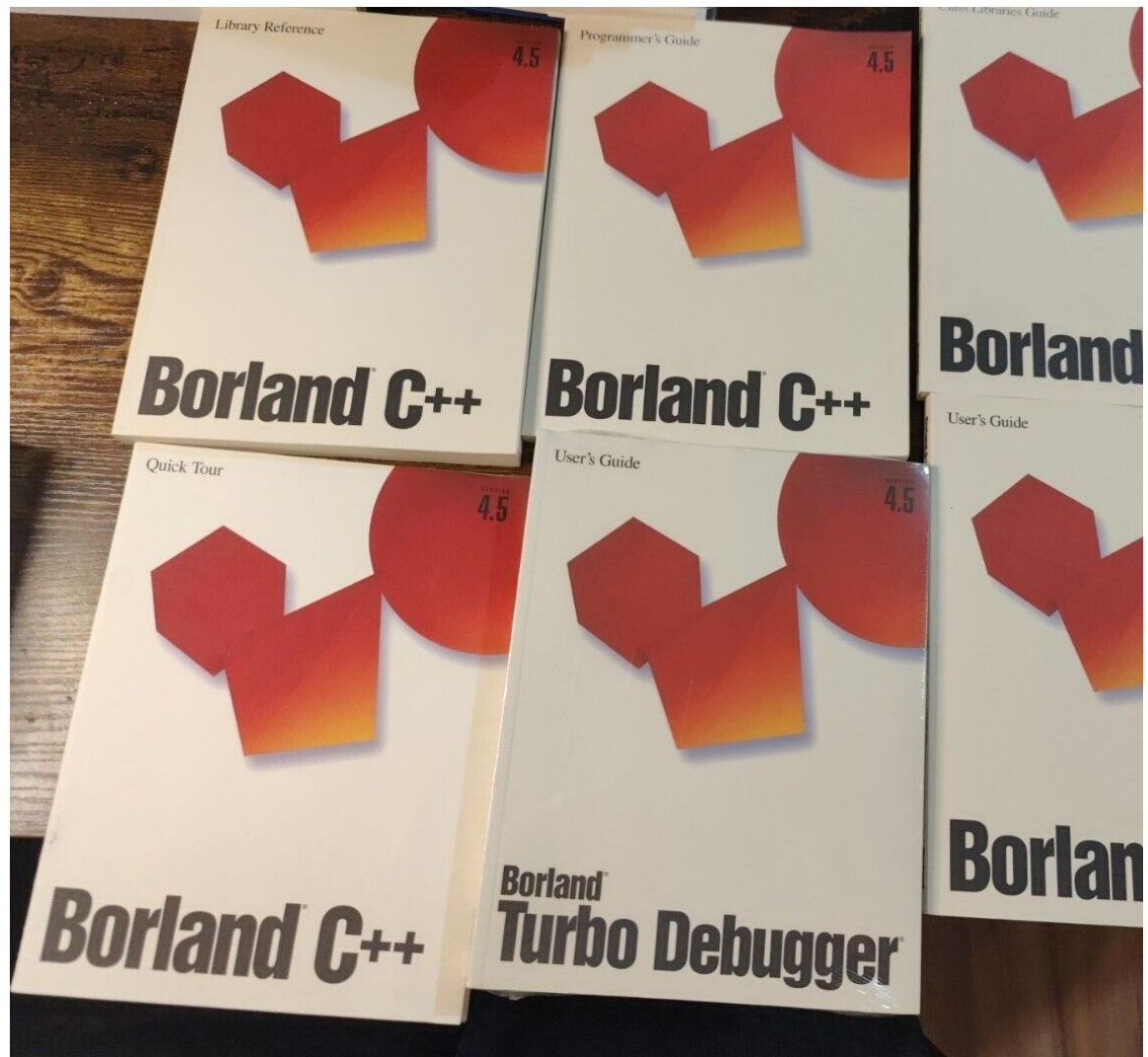
- Built my first software project for sale
- Sold a "dice" I built from scratch game on America Online



### C/C++ Age 15

- Built my own games and software
- Started working with my own databases



**Experimentation is key**

- I built things to use myself (dog fooding)
- I persistently worked on things until I figured it out
- I explored different ways to solve problems

**So what about my kid?**

- Gauge interest level
- Are they curious about how things work?
- They may not have interest...that's ok!

**They have interest, what next?**

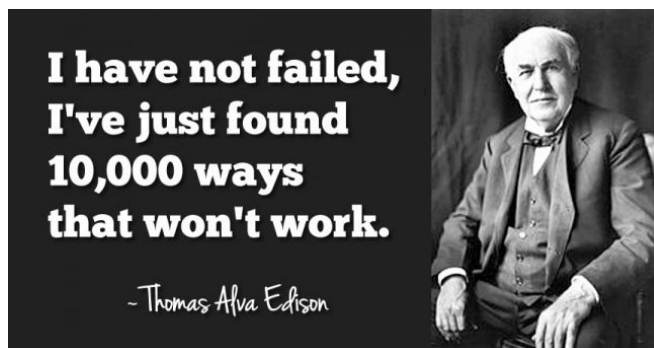
- Find out what they like about computers?
  - Hardware/Software
- Age and comprehension (reading level, math)
- Consider how they learn best (we'll talk more about this)

**High level math? Necessary?**

- NO!
- Can depend on the age and what they want to do.
- For doing graphical/games (sometimes)
- Algebra is a plus

**What does matter?**

- It boils down to this:
  - Learning how to solve problems
  - Getting your mind into that iterative process of experimenting.
  - Try/Fail feedback loop



#### What about the programming language they should learn?

- Not important
- Languages all end up being very similar in the end.

#### Isn't AI supposed to replace coders and software engineers?

- Not convinced - AI/GPT/LLM is just another tool for us to help the experimentation/discovery process
- Things that already tried to kill programming:
  - Advanced compilers (closer to human language)
  - Rapid Application Development (RAD) tools
  - Visual scripting (i.e. Scratch like development)

#### Can't they learn using YouTube tutorials?

- Well, sort of...
  - If they just watch videos - make sure they are taking time to practice what they are learning.
- Videos are good but lack the experimentation process.

#### Encourage Them

- Experiment
- Break large problems into smaller (abstract details)
- Find a solution? Look for other ways to solve that problem
- Which is best?
- Be creative - programming/coding is a creative outlet

#### Find "project" based material

- Teaches your kid to build real things they can use
- Find things that interest them
  - Building games
  - Hacking on Minecraft
  - Pokemon card database, etc...
- Hardware ideas
  - Raspberry Pi device
  - Arduino microprocessor kits

#### Keep it Going

##### Throw interesting problems at them

- Can you make something to catalog or keep track of your school work
- Automating things around the house
- Setting up a network storage for family members
- Family website

##### Pre-college opportunities

- Raikes School Summer Camp (UNL)
- Nebraska.Code() Conference (July 2024)
- Southeast Community College Career Academy (11th & 12th grade)

##### Career Opportunities

- Specialize (look at different areas in computing)
- Look for internships at local business

#### Recommended Resources

##### What to look for in a Resource

- Look for resources that provide that "exploratory" model.

##### Pre-teen

- Realm of Racket (Dr. Racket)
- Turtle Academy (<https://turtleacademy.com/>)
- Scratch (<https://scratch.mit.edu>)
- 30 Days Lost in Space

##### Teen

- How to Design Programs (version 2)
- Minecraft Education
- Nand to Tetris
- Mobile app development
  - Apple iOS Swift Playgrounds
  - Android App Inventor

**More advanced**

- Learn "C"
  - Book: C Programming a Modern Approach (K.N. King)
- Protohackers (design network servers)
- CodeCrafters
- Dave Plummer (Dave's Garage) YouTube

**Wrap Up**

**We looked at**

- Getting Started
- Keeping it Going
- Resources

**Questions?**

**Thank you!!!**

- @tylertorten (X/Twitter)

**Links**

