

Reflection Report on ChatGPT Usage

Cade Aihara

Introduction:

This report about ChatGPT usage follows the lab assignment on ChatGPT. We were tasked with exploring how this AI software works and how it can be used in a constructive way. This report goes into depth on the tasks we were given as well as our commentary on the subject.

Task 1: Familiarizing with ChatGPT:

This task primarily involved reading and playing with ChatGPT. As someone who has never used this software before, I was surprised to see how powerful this tool really is. I got carried away and started looking into how powerful it was because of how it grabbed my interest.

Task 2.1: Code Refactoring:

For this task I chose to use one of my old CS135 assignments. I figured this was a good piece of code to use because of the simplicity of the function of the code and how inexperienced I was as a coder back then. I am aware that in my beginning stages of coding I was very inefficient in the way I coded and thought this would be a great opportunity to see how my code could be refactored.

When giving my code to ChatGPT and telling it to refactor it, I was impressed to see that it cut it down to half its length while maintaining the exact same functionality. I could see myself using this tool in the future as it may see areas of improvement quickly and making me a larger asset in the workforce. The code below is not meant to be read. It is meant to show the difference from original to refactored.

```

C:\Users\Case > cd c:\7473group > cd c:\ahars_out07 > @ cd ahars_REFACTOR07Iss1.cpp > ...
1 #include <iostream>
2 #include <cmath> //allows us to manipulate values
3 #include <math> //allows us to use math functions
4 using namespace std;
5 int main() {
6     double crates, crate, crates1, spaces, space0, space1; //introducing variables (lypis)
7     double crates0, crates1, crates, space0, space1, total; //introducing formula to be used
8
9     cout<<"***** CRATES *****"<<endl;
10    cout<<"Enter length of shipping crate"<<endl; //asking user for dimensions
11    crate<<" ";
12    cin>>crates1;
13    cout<<endl;
14
15    cout<<"Enter width of shipping crate"<<endl;
16    crate<<" ";
17    cin>>crates1;
18    cout<<endl;
19
20    cout<<"Enter height of shipping crate"<<endl;
21    crate<<" ";
22    cin>>crates1;
23    cout<<endl;
24
25    crates0= crates * crate1 * crate1; //assigning math functions to variables
26    crates1 = 2 * (crates1 * crates1 + crates1 * crates1 + crates1 * crates1);
27    crates1 = sqrt(pow(crates1, 2) + pow(crates1, 2) + pow(crates1, 2));
28
29    cout<<"***** SPACE *****"<<endl;
30    cout<<"Enter length of storage space"<<endl; //asking for more dimensions
31    crate<<" ";
32    cin>>space0;
33    cout<<endl;
34
35    cout<<"Enter width of storage space"<<endl;
36    crate<<" ";
37    cin>>space0;
38    cout<<endl;
39
40    cout<<"Enter height of storage space"<<endl;
41    crate<<" ";
42    cin>>space0;
43    cout<<endl;
44
45    space0= space1 * space1 * space1; //assigning math functions to more variables
46    space1 = 2 * (space1 * space1 + space1 * space1 + space1 * space1);
47    space1 = sqrt(pow(space1, 2) + pow(space1, 2) + pow(space1, 2));
48    total = floor(space1 / crates1) * floor(space1 / crates1) * floor(space1 / crates1);
49
50    cout<<"***** RESULTS *****"<<endl;
51    cout<<"Type | Length | Width | Height | Volume | Surface Area | Diagonal |" <<endl;
52    cout<<"-----|-----|-----|-----|-----|-----|-----|" <<endl;
53    cout<<"Crates | ";
54    cout<<endl;
55    cout<<endl;
56    cout<<endl;
57    cout<<endl;
58    cout<<endl;
59    cout<<endl;
60    cout<<endl;
61    cout<<endl;
62    cout<<endl;
63    cout<<endl;
64    cout<<endl;
65    cout<<endl;
66    cout<<endl;
67    cout<<endl;
68    cout<<endl;
69    cout<<endl;
70    cout<<endl;
71    cout<<endl;
72    cout<<endl;
73    cout<<endl;
74    cout<<endl;
75    cout<<endl;
76    cout<<endl;
77    cout<<endl;
78    cout<<endl;
79    cout<<endl;
80    cout<<endl;
81    cout<<endl;
82    cout<<endl;
83    cout<<endl;
84    cout<<endl;
85    cout<<endl;
86    cout<<endl;
87    cout<<endl;
88    cout<<endl;
89    cout<<endl;
90    cout<<endl;
91    cout<<endl;
92    cout<<endl;
93    cout<<endl;
94    cout<<endl;
95    cout<<endl;
96    cout<<endl;
97    cout<<endl;
98    cout<<endl;
99    cout<<endl;
100   cout<<endl;
101   cout<<endl;
102   cout<<endl;
103   cout<<endl;
104   cout<<endl;
105   cout<<endl;
106   cout<<endl;
107   cout<<endl;
108   cout<<endl;
109   cout<<endl;
110   cout<<endl;
111   cout<<endl;
112   cout<<endl;
113   cout<<endl;
114   cout<<endl;
115   cout<<endl;
116   cout<<endl;
117   cout<<endl;
118   cout<<endl;
119   cout<<endl;
120   cout<<endl;
121   cout<<endl;
122   cout<<endl;
123   cout<<endl;
124   cout<<endl;
125   cout<<endl;
126   cout<<endl;
127   cout<<endl;
128   cout<<endl;
129   cout<<endl;
130   cout<<endl;
131   cout<<endl;
132   cout<<endl;
133   cout<<endl;
134   cout<<endl;
135   cout<<endl;
136   cout<<endl;
137   cout<<endl;
138   cout<<endl;
139   cout<<endl;
140   cout<<endl;
141   cout<<endl;
142   cout<<endl;
143   cout<<endl;
144   cout<<endl;
145   cout<<endl;
146   cout<<endl;
147   cout<<endl;
148   cout<<endl;
149   cout<<endl;
150   cout<<endl;
151   cout<<endl;
152   cout<<endl;
153   cout<<endl;
154   cout<<endl;
155   cout<<endl;
156   cout<<endl;
157   cout<<endl;
158   cout<<endl;
159   cout<<endl;
160   cout<<endl;
161   cout<<endl;
162   cout<<endl;
163   cout<<endl;
164   cout<<endl;
165   cout<<endl;
166   cout<<endl;
167   cout<<endl;
168   cout<<endl;
169   cout<<endl;
170   cout<<endl;
171   cout<<endl;
172   cout<<endl;
173   cout<<endl;
174   cout<<endl;
175   cout<<endl;
176   cout<<endl;
177   cout<<endl;
178   cout<<endl;
179   cout<<endl;
180   cout<<endl;
181   cout<<endl;
182   cout<<endl;
183   cout<<endl;
184   cout<<endl;
185   cout<<endl;
186   cout<<endl;
187   cout<<endl;
188   cout<<endl;
189   cout<<endl;
190   cout<<endl;
191   cout<<endl;
192   cout<<endl;
193   cout<<endl;
194   cout<<endl;
195   cout<<endl;
196   cout<<endl;
197   cout<<endl;
198   cout<<endl;
199   cout<<endl;
200   cout<<endl;
201   cout<<endl;
202   cout<<endl;
203   cout<<endl;
204   cout<<endl;
205   cout<<endl;
206   cout<<endl;
207   cout<<endl;
208   cout<<endl;
209   cout<<endl;
210   cout<<endl;
211   cout<<endl;
212   cout<<endl;
213   cout<<endl;
214   cout<<endl;
215   cout<<endl;
216   cout<<endl;
217   cout<<endl;
218   cout<<endl;
219   cout<<endl;
220   cout<<endl;
221   cout<<endl;
222   cout<<endl;
223   cout<<endl;
224   cout<<endl;
225   cout<<endl;
226   cout<<endl;
227   cout<<endl;
228   cout<<endl;
229   cout<<endl;
230   cout<<endl;
231   cout<<endl;
232   cout<<endl;
233   cout<<endl;
234   cout<<endl;
235   cout<<endl;
236   cout<<endl;
237   cout<<endl;
238   cout<<endl;
239   cout<<endl;
240   cout<<endl;
241   cout<<endl;
242   cout<<endl;
243   cout<<endl;
244   cout<<endl;
245   cout<<endl;
246   cout<<endl;
247   cout<<endl;
248   cout<<endl;
249   cout<<endl;
250   cout<<endl;
251   cout<<endl;
252   cout<<endl;
253   cout<<endl;
254   cout<<endl;
255   cout<<endl;
256   cout<<endl;
257   cout<<endl;
258   cout<<endl;
259   cout<<endl;
260   cout<<endl;
261   cout<<endl;
262   cout<<endl;
263   cout<<endl;
264   cout<<endl;
265   cout<<endl;
266   cout<<endl;
267   cout<<endl;
268   cout<<endl;
269   cout<<endl;
270   cout<<endl;
271   cout<<endl;
272   cout<<endl;
273   cout<<endl;
274   cout<<endl;
275   cout<<endl;
276   cout<<endl;
277   cout<<endl;
278   cout<<endl;
279   cout<<endl;
280   cout<<endl;
281   cout<<endl;
282   cout<<endl;
283   cout<<endl;
284   cout<<endl;
285   cout<<endl;
286   cout<<endl;
287   cout<<endl;
288   cout<<endl;
289   cout<<endl;
290   cout<<endl;
291   cout<<endl;
292   cout<<endl;
293   cout<<endl;
294   cout<<endl;
295   cout<<endl;
296   cout<<endl;
297   cout<<endl;
298   cout<<endl;
299   cout<<endl;
300   cout<<endl;
301   cout<<endl;
302   cout<<endl;
303   cout<<endl;
304   cout<<endl;
305   cout<<endl;
306   cout<<endl;
307   cout<<endl;
308   cout<<endl;
309   cout<<endl;
310   cout<<endl;
311   cout<<endl;
312   cout<<endl;
313   cout<<endl;
314   cout<<endl;
315   cout<<endl;
316   cout<<endl;
317   cout<<endl;
318   cout<<endl;
319   cout<<endl;
320   cout<<endl;
321   cout<<endl;
322   cout<<endl;
323   cout<<endl;
324   cout<<endl;
325   cout<&lt
```

Task 2.2: Documentation Assistance:

For documentation assistance, I chose to build off of the refactored code from the previous task. For this task I prompted ChatGPT requesting documentation on my code to increase reader's ability to understand my code. It seemed to work perfectly and was able to move on rather quickly. Here is what the code looked like after documentation.

```

C: > Users > Cade > cs472-group4 > chatgpt > Cade_Aihara > Cade_Aihara_documented2.2.cpp > ...
1  #include <iostream> // Standard input/output stream
2  #include <iomanip> // Input/output manipulators
3  #include <cmath> // Math functions
4  using namespace std;
5
6  int main() {
7      // Variables for dimensions of shipping crate and storage space
8      double crateL, crateW, crateH, spaceL, spaceW, spaceH;
9
10     // Variables for volume, surface area, and diagonal of crate and space
11     double crateVOL, crateSA, crateD, spaceVOL, spaceSA, spaceD, total;
12
13     // Prompting user for dimensions of the shipping crate
14     cout << "SHIPPING CRATE" << endl;
15     cout << "Enter length, width, and height of the shipping crate: ";
16     cin >> crateL >> crateW >> crateH;
17
18     // Calculating volume, surface area, and diagonal of the shipping crate
19     crateVOL = crateL * crateW * crateH;
20     crateSA = 2 * (crateL * crateW + crateL * crateH + crateW * crateH);
21     crateD = sqrt(pow(crateL, 2) + pow(crateW, 2) + pow(crateH, 2));
22
23     // Prompting user for dimensions of the storage space
24     cout << "STORAGE SPACE" << endl;
25     cout << "Enter length, width, and height of the storage space: ";
26     cin >> spaceL >> spaceW >> spaceH;
27
28     // Calculating volume, surface area, and diagonal of the storage space
29     spaceVOL = spaceL * spaceW * spaceH;
30     spaceSA = 2 * (spaceL * spaceW + spaceL * spaceH + spaceW * spaceH);
31     spaceD = sqrt(pow(spaceL, 2) + pow(spaceW, 2) + pow(spaceH, 2));
32
33     // Calculating total number of crates that can fit in the storage space
34     total = floor(spaceL / crateL) * floor(spaceW / crateW) * floor(spaceH / crateH);
35
36     // Outputting the table header
37     cout << "+-----+-----+-----+-----+-----+-----+-----+" << endl;
38     cout << "| Type   | Length | Width  | Height | Volume   | Surface Area | Diagonal |" << endl;
39     cout << "+-----+-----+-----+-----+-----+-----+-----+" << endl;
40
41     // Outputting data for the shipping crate
42     cout << "| Crate | " << setw(7) << fixed << setprecision(1) << crateL << " | "
43     << setw(7) << crateW << " | " << setw(7) << crateH << " | "
44     << setw(11) << crateVOL << " | " << setw(13) << crateSA << " | " << setw(9) << crateD << " |" << endl;
45
46     // Outputting data for the storage space

```

Task 2.3: Understanding Complex Code:

This task of understanding complex code was a bit challenging. I had to rephrase my prompt when asking ChatGPT to do what I requested. It would often make comments on the code after it sent me the code back. This made the comments independent from the file which was undesired.

After playing around with the wording and phrasing of my request I was able to finally receive what I needed. It integrated the comments into the code and was happy with the result. My code was not very complex so there wasn't a ton to break down. However, I was able to explain each of my code lines in detail further than the previous step.

```

C: > Users > Cade > cs472-group4 > chatgpt > Cade_Aihara > G+ cadeAihara_complexCode2_3.cpp > main()
1  #include <iostream> // Including the library for standard input/output stream
2  #include <iomanip> // Including the library for input/output manipulators
3  #include <cmath> // Including the library for math functions
4  using namespace std;
5
6  int main() {
7      double crateL, crateW, crateH, spaceL, spaceW, spaceH; // Declaring variables for dimensions of shipping crate
8      double crateVOL, crateSA, crateD, spaceVOL, spaceSA, spaceD, total; // Declaring variables for volume, surface area, diagonal, and total
9
10     cout << "SHIPPING CRATE" << endl; // Prompting user for dimensions of the shipping crate
11     cout << "Enter length, width, and height of the shipping crate: ";
12     cin >> crateL >> crateW >> crateH; // Taking input for crate dimensions
13
14     crateVOL = crateL * crateW * crateH; // Calculating volume of the crate
15     crateSA = 2 * (crateL * crateW + crateL * crateH + crateW * crateH); // Calculating surface area of the crate
16     crateD = sqrt(pow(crateL, 2) + pow(crateW, 2) + pow(crateH, 2)); // Calculating diagonal of the crate
17
18     cout << "STORAGE SPACE" << endl; // Prompting user for dimensions of the storage space
19     cout << "Enter length, width, and height of the storage space: ";
20     cin >> spaceL >> spaceW >> spaceH; // Taking input for storage space dimensions
21
22     spaceVOL = spaceL * spaceW * spaceH; // Calculating volume of the storage space
23     spaceSA = 2 * (spaceL * spaceW + spaceL * spaceH + spaceW * spaceH); // Calculating surface area of the storage space
24     spaceD = sqrt(pow(spaceL, 2) + pow(spaceW, 2) + pow(spaceH, 2)); // Calculating diagonal of the storage space
25
26     total = floor(spaceL / crateL) * floor(spaceW / crateW) * floor(spaceH / crateH); // Calculating total number of crates
27
28     cout << "+-----+-----+-----+-----+-----+-----+-----+" << endl; // Outputting separator
29     cout << "| Type   | Length | Width  | Height | Volume   | Surface Area | Diagonal |" << endl;
30     cout << "+-----+-----+-----+-----+-----+-----+-----+" << endl;
31
32     cout << "| Crate | " << setw(7) << fixed << setprecision(1) << crateL << " | " // Outputting data for the shipping crate
33     << setw(7) << crateW << " | " << setw(7) << crateH << " | "
34     << setw(11) << crateVOL << " | " << setw(13) << crateSA << " | " << setw(9) << crateD << " |" << endl;
35
36     cout << "| Space | " << setw(7) << spaceL << " | " << setw(7) << spaceW << " | " // Outputting data for the storage space
37     << setw(7) << spaceH << " | " << setw(11) << spaceVOL << " | "
38     << setw(13) << spaceSA << " | " << setw(9) << spaceD << " |" << endl;
39
40     cout << "+-----+-----+-----+-----+-----+-----+-----+" << endl; // Outputting separator
41
42     cout << fixed << setprecision(1) << total << " crates can fit in the storage space." << endl; // Outputting total

```

Task 2.4: Workflow Automation with GitHub Actions:

This task was also rather simple. I asked it to generate a .yaml file to test my code for potential issues it may encounter if it was deployed. This was a ton easier than making the yaml file myself so I thought this was a really cool and helpful feature.

```

name: Shipping Crate CI Workflow
on:
  push:
    branches:
      - main
  pull_request:
    branches:
      - main
jobs:
  build:
    runs-on: ubuntu-latest
    container: python:3.9-slim
    steps:
      - name: Checkout
        uses: actions/checkout@v3
      - name: Install dependencies
        run: |
          python -m pip install --upgrade pip
          pip install -r requirements.txt
      - name: Lint with Flake8
        run: |
          flake8 src --count --select=E9,F63,F7,F82 --show-source --statistics
          flake8 src --count --max-complexity=10 --max-line-length=127 --statistics
      - name: Run unit tests with nose
        run: nosetests -v --with-spec --spec-color --with-coverage --cover-package=src
      - name: SonarQube analysis
        run: |
          # Replace <SONAR_TOKEN> and <SONAR_PROJECT_KEY> with your SonarQube token and project key
          sonar-scanner \
            -Dsonar.projectKey=<SONAR_PROJECT_KEY> \
            -Dsonar.sources=src \
            -Dsonar.host.url=<SONARQUBE_URL> \
            -Dsonar.login=<SONAR_TOKEN>

```

Impact:

The impact of this assignment was huge. I personally have never used ChatGPT so this was kind of mind blowing to me. This software has the ability to help me understand software development better than I do now. I am able to submit any piece of code and get further clarification on how it works and how to make it better. I believe this is a great tool for learning as long as it is not abused.

Lessons Learned:

As previously mentioned I learned a ton from this. I learned how helpful this tool can be. If you are precise enough with your words you will be able to do a lot with this. I learned that wording really matters in this tool and that it is really powerful. You can make it do work for you and clear up time for other tasks (if permitted). Overall I learned that AI is not always bad and can be used for constructive purposes.

Future Applications:

After this lab I can definitely see myself using this in the future. I can see this being deployed in the work force. Unfortunately I can see a lot of jobs being replaced by AI but it may

boost productivity of the human civilization. This is a little off topic but like how the industrial revolution happened, I can see the same thing happening with AI. We will be able to solve greater problems if we free up time by completing tasks with AI.

Conclusion:

Initially, I was hesitant to use ChatGPT. It has been around for a bit now but this is my first time actually using it. I have heard negative things like it being used for cheating on assignments and I wanted to stay clear from that to avoid any accusations.

After this task I was able to see that this can be a very helpful tool. As long as it is not abused and used with integrity, this can be beneficial to students and those in the workforce.

This impacted me because this displayed a new tool to use in the software development area. It shows that instead of spending time optimizing code or making comments or test files, you can make these things with AI and move on to bigger problems.

As I mentioned before, I learned a lot of lessons from this lab. It showed me a great tool I can use to boost productivity. I can also definitely see myself using this in the future. Even if I decide not to pursue a career in computer science or software development, I can use this anywhere. To improve resumes, emails, and much more. This can be used by anyone.

In conclusion, ChatGPT is a great piece of software that can be very effective in freeing up time in tasks that actually require human brain and creativity. It can be used for the simple tasks like adding comments, optimizing code, or even making test files. You can spend your time elsewhere maybe coming up with another idea. This is a great tool that has great power unless it is abused.