

When not otherwise specified, blocks of code provided for a given question are assumed to be the sole contents of the `int main()` function for a new program specifically written for that question (and ending with `return 0;`). For example, if I give you a block of code:

```
string a = "Fred";  
cout << "Hello" << a;
```

I am really asking you to consider a program:

```
#include<iostream>  
using namespace std;  
  
int main() {  
    string a = "Fred";  
    cout << "Hello" << a;  
    return 0;  
}
```

What is the output of this code?

```
int a=3, b;  
float c;  
c=2.1;  
b=c*a;  
cout << b;
```

Output is: **6** ($2.1 \times 3 = 6.3$... rounded to nearest integer)

Let us consider the following code:

```
int g;  
cin >> g;  
if(!(g<=5))  
    cout << "Output 1";  
else  
    cout << "Output 2";
```

Describe the type (or types) of values for `g` that cause this code to display `Output 1`

Re-write the above code with proper indentation.

Re-write the above code, removing the `!` symbol.

```
int g;  
cin >> g;  
if(g>5)  
    cout << "Output 1";  
else  
    cout << "Output 2";
```

Let us consider the following code:

```
int d=6%4;
float m=3.2;
m /= d;
cout >> d >> endl;
cout >> m;
```

Identify any compiler errors in the code.

Presuming the compiler errors are fixed (if there are any), what would be the output of this code?

Presuming the compiler errors are fixed (if there are any), what would be the output if the third line said

`m = d+2;` rather than `m /=d;` ?

2
4

Let us consider the following code:

```
int a,b,c;
a=4; b=6; c=a-b; b=12; a==4;
cout << a << "+" << b << "=" <<c;
```

Which statement declares the variable b?

Which statement initializes the variable a?

What is the output of this code?

4+12=-2

Let us consider the following code, to output a greeting to the screen multiple times:

```
int a,b;
a=4;
while(a>2);
{cout << Hello world;
a--;}
```

Identify any errors in the code above, and how to fix them. For each error (there may be only one error, or no errors!), say if it is a compile error or a run-time error.

Rewrite the code with proper indenting.

Write a `do-while` loop that will have the same output as the one above, presuming any errors are fixed first.

YOU WILL NOT BE TESTED ON DO-WHILE LOOPS ON THE MIDTERM!!

There are a variety of proper answers for this question. The simplest one is:

```
int a,b;
a=4;
do
{cout << "Hello world";
a--;}
while(a>2);
```

Let us consider an example run of the following program. Note, text in red underline is user input.

```
>./myProgram
```

```
Give me a number: 1234
```

```
1234
```

```
>
```

Write the code (just the inside of the `int main()` function) for `myProgram.cpp`.

Provide three different variable types that can be used to hold the user input.

int, float, and string

What is an escape sequence that can be used in your code for `myProgram.cpp` ?

Let us consider example runs of the following program. It takes a positive integer number input `n` and outputs the result of multiplying that number by itself `n` times. For example, if a user gives the number 3, the output will be the result of $3 \times 3 \times 3$, which is 27. Note, text in red underline is user input.

```
>./myProgram
```

```
Give me a number: 2
```

```
4
```

```
>./myProgram
```

```
Give me a number: 3
```

```
27
```

```
>./myProgram
```

```
Give me a number: 4
```

```
256
```

```
>
```

Presuming we do not check whether the input number is positive, write the code for `myProgram.cpp`.

```
int num, product, count;
cout << "Give me a number: ";
cin >> num;
product=1; count=1;
while(count<=num){
    product=product*num;
    count++;
}
cout << product;
```

Write the condition for an `if` statement to determine if the input number is positive.

Write a test for the input number being positive that can be used in a `switch` statement.

Consider the following code:

```
char d='e', e='c', f='n';
if(d=='e')
    f=d;
    cout << "d and e are equal" << endl;
f=f--;
cout << f;
```

What is the output of the above code?

What would the output of the code be if the condition for the `if` statement was `d=e`?

We want to create a variable indicating whether someone is old enough to vote in the country they live in. Countries are represented by single letters. In Canada (represented by C), the voting age is 18. In Nicaragua (represented by N), the voting age is 16. In Singapore (represented by S), the voting age is 21. Presume we have a variable for country and a variable for age.

What type should the country variable be?

`char`

What type should the age variable be?

`int` (alternatively, can be `float`)

How can we define the old-enough-to-vote variable in one line (based on the age and country variables)?

Let us consider example runs of the following program. It takes a number as input and counts up three times by 4.

```
> ./myProgram
Enter starting number: 3
3 7 11 15
> ./myProgram
Enter starting number: -2.5
-2.5 1.5 5.5 9.5
>
```

Write the code for `myProgram.cpp`.

Consider the following code:

```
int age;
char freePass;
cout << "What is your age? ";
cin >> age;
cout << "Do you have a free pass? ";
cin >> freePass;
if(age>=18||age<80&&freePass!='N') {
int age;
cout << "You get in for free" << endl;
age = 20;
}
cout << "You are " << age << " years old\n";
```

Under what conditions will the message “You get in for free” appear?

If age is greater than or equal to 18, or if age is less than 18 and freePass is any character except 'N'

Presuming the “You get in for free” message appears, what number will be displayed in the “You are ??? years old” message?