

Conditions and Logic II

1. What is the output of the following program? Draw the control flow chart. Provide different test values in order to get different outcomes.

```
1 def mystery(people: int, bikes: int, cars: int) -> None:
2     if (bikes + cars > people):
3         print("no need of walking...")
4         if cars > bikes:
5             print("too many cars")
6         else:
7             print("bikes may work")
8     if bikes > people:
9         print("bikes are cool")
10    else:
11        print("let's stay home then")
12 def main() -> None:
13     mystery(30, 28, 10) # test 1
14 main()
```

2. What is the output of the following program? Draw the control flow chart. Provide different test values in order to get different outcomes.

```
1 def mystery(students: int, desks: int, professors: int) -> None:
2     if (students + professors) > desks:
3         print("There are not enough desks!")
4     else:
5         print("There are enough desks for everyone to have their own.")
6         leftover = desks - (students + professors)
7         if leftover >= students:
8             print("There will be A LOT of empty desks.")
9         elif leftover >= professors:
10            print("There will be exactly", leftover, "empty desks.")
11        elif leftover >= 1:
12            print("There will be at least one empty desk.")
13        else:
14            print("There will not be any empty desks." )
15
```

```
16 def main() -> None:
17     mystery(28, 100, 5) # test 1
18 main()
```

3. What is the output of the following program? Draw the control flow chart. Provide different test values in order to get different outcomes.

```
1 def mystery(temperature: int, thickness: float) -> None:
2     if (temperature >= -9 and temperature <= -5):
3         print(temperature, "C is good for speed skating")
4         if (thickness < 2.5):
5             more = 2.5 - thickness
6             print("The ice needs to be", more, "cm thicker")
7         else:
8             print(thickness, "cm is thick enough ice")
9             print("According to Mark Messer")
10    elif (temperature < -3):
11        print(temperature, "C is good for figure skating")
12        if (thickness >= 4.5 and thickness <= 5):
13            print(thickness, "cm ice is good")
14    else:
15        print("The ice is all wrong")
16 def main() -> None:
17     mystery(-7, 2.0) #test 1
18 main()
```

4. Rewrite the program below to have only a single if statement. Your solution must have exactly one `if`, zero or more `elif` branches, and zero or one `else` branch. (In your solution, you need only rewrite the part below the comment.) Draw the control flow chart and use it in the simplification.

```
1 def main() -> None:
2     response = input("Is it raining out? ")
3     rainy = response == 'yes'
4     wind = int(input("What is the wind speed? "))
5     # rewrite the program from here down...
6     if rainy:
7         if wind > 30:
8             print ("Wear your rain slicker!")
9         else:
10            print ("Bring your umbrella!")
11    else:
12        if wind > 30:
13            print ("Wear your windbreaker!")
14    main()
```

5. Rewrite the program below to have only a single if statement. Your solution must have exactly one `if`, zero or more `elif` branches, and zero or one `else` branch. (In your solution, you need only rewrite the part below the comment.) Draw the control flow chart and use it in the simplification.

```
1 def main() -> None:
2     response = input("Is it raining out? ")
3     rainy = response == 'yes'
4     month = int(input("Enter the month: "))
5     A = "Ride your snowmobile to class."
6     B = "Take the shuttle."
7     C = "Ride your bike to class."
8
9     # rewrite the program from here down...
10    if month > 10:
11        print(A)
12    if month < 3:
13        print(A)
14    else:
15        if rainy:
16            print(B)
17        else:
18            print(C)
19    main()
```