

CSC1015F Assignment 5

Control (if, for, while)

Assignment Instructions

This assignment involves constructing Python programs that use input and output statements, 'if' and 'if-else' control flow statements, 'while' statements, 'for' statements, and statements that perform numerical manipulation.

NOTE Your solutions to this assignment will be evaluated for correctness and for the following qualities:

- Documentation
 - Use of comments at the top of your code to identify program purpose, author and date.
 - Use of comments within your code to explain each non-obvious functional unit of code.
- General style/readability
 - The use of meaningful names for variables and functions.
- Algorithmic qualities
 - Efficiency, simplicity

These criteria will be manually assessed by a tutor and commented upon. In this assignment, up to 10 marks will be deducted for deficiencies.

Question 1 [30 marks]

Write a program called 'grid.py' that accepts a number, n , where $-6 < n < 2$. The program will print out the numbers n to $n+41$ as 6 rows of 7 numbers. The first row will contain the values n to $n+6$, the second, the values $n+7$ to $n+7+6$, and so on. Your program should cater for conditions whereby the user has entered values less than -6 and/or greater than 2. Note that -6 and 2 are not included (see the sample IO overleaf).

Numbers are printed using a field width of 2, and are right-justified.

Fields are separated by a single space. There are no spaces after the final field.

Sample IO (The input from the user is shown in **bold** font – do not program this):

Enter the start number:

-2

```
-2 -1  0  1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31 32
33 34 35 36 37 38 39
```

Sample IO (The input from the user is shown in **bold font** – do not program this):

Enter a number between -6 and 2:

2

Invalid input! The value of 'n' should be between -6 and 2.

HINT: Use a 'for' loop within a 'for' loop.

Question 2 [40 marks]

Write a program called 'printmonth.py' that asks the user for a month name and start day and then prints the calendar for that month in a 6 row by 7 column grid. (Ignoring issues of leap years, assume February has 28 days). Your calendar should cater for conditions whereby the user has entered an incorrect name of a month and/or a start date (*See the sample IO below*).

Sample IO (The input from the user is shown in **bold font** – do not program this):

Enter the name of a month(January, ..., December):

September

Enter the start day (1 for Monday, ..., 7 for Sunday):

4

Mo	Tu	We	Th	Fr	Sa	Su
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

Sample IO (The input from the user is shown in **bold font** – do not program this):

Enter the name of a month (e.g. January, ..., December):

June

Enter the start day (1 for Monday, ..., 7 for Sunday):

9

Invalid calendar: you have either entered an incorrect month name or start day.

Sample IO (The input from the user is shown in **bold font** – do not program this):

Enter the name of a month (e.g. January, ..., December):

Month

Enter the start day (1 for Monday, ..., 7 for Sunday):

3

Invalid calendar: you have either entered an incorrect month name or start day.

Sample IO (The input from the user is shown in **bold font** – do not program this):

Enter the name of a month (e.g. January, ..., December):

July

Enter the start day (1 for Monday, ..., 7 for Sunday):

2

Mo	Tu	We	Th	Fr	Sa	Su
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Note (for the sake of automatic marking) that output must always form a 6 row by 7 column grid. Spaces should be used where necessary.

HINT: Calculate the start value based on the day, print spaces for numbers less than one and for numbers greater than the number of days in the month.

Question 3 [30 marks]

In many user interfaces a user is presented with a set of options, one of which is selected to perform a particular task. Text-based programs often print out these options linearly and allow users to select options continuously until a specific task has been chosen.

In the precursor to the Internet, public Bulletin Board Systems (BBS) were used to exchange information (messages/files) in a community of users. Write a program called 'bulletin.py' to simulate a simple BBS with one stored message and 2 fixed files, as indicated in the output. Study the output carefully before you start programming!

Use a default message of "no message yet"; and if a file cannot be located, output "File not found".

Sample IO (The input from the user is shown in **bold font** – do not program this):

Welcome to UCT BBS

MENU

(E)nter a message

(V)iew message

(L)ist files

(D)isplay file

e(X)it

Enter your selection:

E

Enter the message:

test message

Welcome to UCT BBS

MENU

(E)nter a message

(V)iew message

(L)ist files

(D)isplay file

e(X)it

Enter your selection:

V

The message is: test message

Welcome to UCT BBS

MENU

(E)nter a message

(V)iew message

(L)ist files

(D)isplay file

e(X)it

Enter your selection:

X

Goodbye!

Sample IO (The input from the user is shown in **bold font** – do not program this):

Welcome to UCT BBS

MENU

(E)nter a message

(V)iew message

(L)ist files

(D)isplay file

e(X)it

Enter your selection:

L

List of files: 42.txt, 1015.txt

Welcome to UCT BBS

MENU

(E)nter a message

(V)iew message

(L)ist files

```
(D)isplay file
e(X)it
Enter your selection:
D
Enter the filename:
42.txt
The meaning of life is blah blah blah ...
Welcome to UCT BBS
MENU
(E)nter a message
(V)iew message
(L)ist files
(D)isplay file
e(X)it
Enter your selection:
D
Enter the filename:
1015.txt
Computer Science class notes ... simplified
Do all work
Pass course
Be happy

Welcome to UCT BBS
MENU
(E)nter a message
(V)iew message
(L)ist files
(D)isplay file
e(X)it
Enter your selection:
D
Enter the filename:
1016.txt
File not found
Welcome to UCT BBS
MENU
(E)nter a message
(V)iew message
```

```
(L)ist files
(D)isplay file
e(X)it
Enter your selection:
x
Goodbye!
```

Submission

Create and submit a Zip file called 'ABCXYZ123.zip' (where ABCXYZ123 is YOUR student number) containing `grid.py`, `printmonth.py` and `bulletin.py`.

NOTES:

1. FOLDERS ARE NOT ALLOWED IN THE ZIP FILE.
2. As you will submit your assignment to the Automarker, the Assignment tab may say something like "Not Complete". THIS IS COMPLETELY NORMAL. IGNORE IT.