

Quiz #1

CISC 121, Winter 2011

*****Please do NOT turn over this page and begin the quiz until you are given permission*****

STUDENT NUMBER: -

Please do not write your name on the quiz, but write your student number clearly above and on every page that contains an answer.

Please remain quiet in the classroom. If you finish early, give your paper to the instructor or put it in the space provided and leave quietly to avoid disturbing those who are still working. Your eyes should always be on your paper, the instructor, or the blackboard. Please do not incur suspicion by looking all around the room, especially not in the direction of other students' papers. **Academic dishonesty of any kind will not be tolerated.**

You may use **one page of notes** (your "summary sheet") as a memory aid during the quiz. You may not refer to any other notes or books you may have brought with you. You may not use a calculator or computer.

Please make sure that you hand in every page of the quiz that contains an answer and that these pages are securely stapled together. There will be a stapler available.

You have **45 minutes** to complete this quiz. The quiz is marked out of **30 points**. Budget your time accordingly. The weight for each question is given with the question.

Please try to answer each question in the space provided. If you write an answer in a different place, please write a note indicating where I should look for it. There is a blank page at the end of the quiz (before the summary sheet). You may use this for extra space or rough work if you need it. You may also use the backs of the pages for extra space.

Your answers will be marked for correctness, not for style. Comments are not required. However, I can't mark anything that isn't legible, and clear style or comments may help determine partial credit.

If you write more than one answer to a question, please make it very clear which answer should be marked. Circle your final answer and write "mark this" or cross out your other answers. If it's not obvious what you intended, I will arbitrarily pick an answer to mark.

This area for marking purposes only:

1. _____ / 10
2. _____ / 10
3. _____ / 10
total: _____ / 30

Question 1 (10 points):

For this question, I have written a Python function. In each place where the function prints something, I have provided a box. Your job is to write the output of the function in the boxes.

If you think that there's an error in one of the sections of code (something that would make Python stop and print an error message), write "error" in the box. But continue with the remaining sections as if the error hadn't happened.

```
def question1():
```

```
    a = 47
```

```
    b = "cats"
```

```
    print(a, b, sep="++")
```

```
    print(13/2)
```

```
    print(13//2)
```

```
    numbers = [1,2,3,5,8,13,21]
```

```
    print(numbers[2])
```

```
    print(numbers[-3])
```

```
    print(numbers[1:5])
```

```
    languages = ["Java", "Python", "C++", "Turing"]
```

```
    print(languages[1][3] + languages[3][5:8])
```

```
// continued on the next page
```

Question 1, continued:

```
print(7 > 3 and 4 < 2)
```

```
for value in [14, 37]:  
    if value < 20:  
        print(1)  
    elif value < 15:  
        print(2)  
    elif value > 30:  
        print(3)  
    else:  
        print(4)  
print(5)
```

Question 2 (10 points):

Write a Python function called `oddProduct`. This function should take no parameters. It must prompt the user for an integer. (Assume the user will type something that may be interpreted as an integer.) If the number the user entered (call it `n`) is zero or negative, the function should write an error message. If `n` is positive, the function must print the product of all the odd numbers between 1 and `n`, inclusive. For example, if `n` is 7, the function should print 105 (because $1*3*5*7 = 105$).

Here is an interactive session using this function:

```
>>> oddProduct()
enter a positive integer: 5
the product is 15
>>> oddProduct()
enter a positive integer: 6
the product is 15
>>> oddProduct()
enter a positive integer: 7
the product is 105
>>> oddProduct()
enter a positive integer: 0
error: input is not positive
>>> oddProduct()
enter a positive integer: -7
error: input is not positive
>>> oddProduct()
enter a positive integer: 1
the product is 1
```

Question 3 (10 points):

Write a Python function called `noRepeats`, which has one parameter. You may assume this parameter will be a string. The function must return a string which is a copy of the parameter except that if a character occurs two or more times in a row the result will replace that sequence of characters with a single instance of the character.

Here is an interactive session using this function:

```
>>> newstr = noRepeats('book')
>>> print(newstr)
bok
>>> newstr = noRepeats('dash -- dot!!!')
>>> print(newstr)
dash - dot!
>>> newstr = noRepeats('no repeated characters')
>>> print(newstr)
no repeated characters
>>> newstr = noRepeats('aaaaaaaaeeeeeeeeeeeeiiiiiiiiiiiiioooooooooouuuuuu')
>>> print(newstr)
aeiou
```

Your `noRepeats` function may not do any input or output. It must use its parameter and return a result.

Student number: _____

(This page left blank for rough work)