Started on	Sunday, 19 March 2023, 10:10 AM
State	Finished
Completed on	Sunday, 19 March 2023, 10:37 AM
Time taken	27 mins 1 sec
Marks	8.00/8.00
Grade	**10.00** out of 10.00 (**100**%)

Question 1

Correct

Mark 1.00 out of 1.00

Consider a situation where you don't have function to calculate power (pow() function in C) and you need to calculate x^n where x can be any number and n is a positive integer. What can be the best possible time complexity of your power function?

a. O(nLogn)

b. O(LogLogn)

c. O(Logn) ✔

d. O(n)

Power of a number can be calculated recursively.

Refer https://www.geeksforgeeks.org/write-a-c-program-to-calculate-powxn/

The correct answer is: O(Logn)

Question 2

Correct

Mark 1.00 out of 1.00

Output of the following program is?

```
void function( int n)
{
   if(n==0)
      return;
   printf("%d ",n*2);
   unction(n-1);
}

int main()
{
   function(100);
   return 0;
}
```

a. 200, 199, 198, 197, …. 1

b. 100, 99, 98, 97, …. 0

c. 100

d. 200, 198, 196, 194, …. 2 ✔

The correct answer is: 200, 198, 196, 194, …. 2

Question 3

Correct

Mark 1.00 out of 1.00

Consider Following Code

```
void my_recursive_function()
{
    my_recursive_function();
}

int main()
{
    my_recursive_function();
    return 0;
}
```

What will happen when the above snippet is executed?

- a. The code will show a compile time error
- b. The code will run for some time and stop when the stack overflows ✔
- c. The code will be executed successfully and no output will be generated
- d. The code will be executed successfully and random output will be generated

The correct answer is: The code will run for some time and stop when the stack overflows

Question 4

Correct

Mark 1.00 out of 1.00

Given below is a function (isNumberEven) that predicts whether a given number is even. The function should return True if a number is even and False otherwise. Fill in the blanks of the return statements:

isNumberEven(N)

```
if N == 0 :
    return [True] ✔
elif N ==1 :
    return [False] ✔
else:
    return [isNumberEven(N-2)] ✔
```

[isNumberEven(N/2)] [isNumberEven(N-1)]

The thread at freecodechamp explains this in detail

https://forum.freecodecamp.org/t/even-odd-number-check-using-the-recursive-function/310067

Question 5

Correct

Mark 1.00 out of 1.00

Select true statement(s) considering Iteration vs Recursion

- [ ] a. Recursion is the best way to solve a problem over iteration
- [x] b. Iteration increases the performance of an algorithm ✔
- [x] c. Considering readability and clearness, iteration is better than recursion ✖
- [x] d. Recursion need more memory than iteration ✔

The correct answers are: Recursion need more memory than iteration, Iteration increases the performance of an algorithm

Question 6

Correct

Mark 1.00 out of 1.00

Algorithm(s) which use divide and conquer approach

- [x] a. Merge sort ✔
- [x] b. Binary search ✔
- [ ] c. Insertion Sort
- [ ] d. Selection Sort

The correct answers are: Binary search, Merge sort

Question 7

Correct

Mark 1.00 out of 1.00

Consider the following function,

```
fn(N)
if N > 100 :
        return N - 10
else:
        return fn(fn(N+11))
```

Select one:

- ( ) a. Mutual Recursion
- ( ) b. Multiple Recursion
- (•) c. Nested Recursion ✔
- ( ) d. Linear Recursion
- ( ) e. Tail Recursion

Your answer is correct.

“In this recursion, a recursive function will pass the parameter as a recursive call. That means recursion inside recursion” - GeeksforGeeks.

Here the function passes itself as a parameter to the recursive call - fn(fn(N+11))

The correct answer is:
Nested Recursion

Question 8

Correct

Mark 1.00 out of 1.00

In a little game, a computer is going to randomly select an integer from 1 to 2000. You'll keep guessing numbers until you find the computer's number, and the computer will tell you each time if your guess was too high or too low. How many quesses you need atmost in your worst case scenario given you use an optimal strategy?

Answer: 11 ✔

The correct answer is: 11

Previous activity
Recursion & Divide and conquer- Take Home Assignment

Jump to...

Next activity
Learning outcomes