Name: Gaurav Taneja
ID: 1001955801

Home Work 3

1) Why should you be interested in learning about Lambda Calculus?
Answer: Can encode any computation, We can encode anything. Basis of functional programming(PASCAL). Lambda calculus is present in new languages. It variables, no side effect, just compute and gives output.

2) How do you encode the concepts of TRUE, FALSE, NOT, AND, OR?
Answer: True function:
lambda x. lambda y. x
False Function:
lambda x. lambda y. y
Not function:
lambda b. b FALSE TRUE
function: (lambda b. b FALSE TRUE) I/P: TRUE
True function: (lambda x. lambda y. x) I/P: FALSE TRUE O/P: FALSE
OR: lambda x. lambda y. x true y
AND: lambda x. lambda y. x y false

3)What is important about the Lambda Calculus expression called 'Y Combinator'?
Answer: Y combinator expression is used to do recursion if there is no recursion provision in the language. Recursion is defining the thing in terms of there selves example of recursion is factorial. By using lambda calculus expression Ycombinator performs recursion. The y combinator encodes the recursion using loop.

4) Write the Y Combinator expression in Lambda Calculus?
The recursion can be done by using loop. Basic definition is below.
loop= (lambda x. x x) (lambda x. x x)
Note: Function is the recursion and ‘x x’ is self application. For the ‘x’ in the first expression input is the function, this will reach the first step again. This is recursion.
Eg:
lambda Func.(lambda x. Func(x x)) (lamda x. Func(x x))

5)Where did the language 'Haskell' get its name?
Answer: The name of the language comes from the Haskell b. curry. He use to work with lambda calculus. The initial name decided by the committee was curry but they decide to use first name that is Haskell.

6) In the video it was mentioned that Erlang was used to code what?
Answer: Elang is example of functional programming. Erlang was used to implement whatsapp.

7) How is 'pattern matching' used?
Answer: Pattern matching is used in functional programming where is there is no side effect. The pattern matching checks with the expression if its a match a particular function will performed or else other function will be performed.

8) Complete this sentence: "NP problems are hard to solve but easy to _____" .
Answer: NP problems are hard to solve but easy to check.

Name: Gaurav Taneja
ID: 1001955801

9) What is the example of an NP problem used in the video?
Answer: Example of an NP problem used in the video is factoring a huge number. Example factoring for 100 will be a big problem.

10)What are the TV shows mentioned in the video?
Answer: 'The Simpsons' and 'Futurama' were mentioned in the video.

11) Floating point numbers are essentially what?
Answer: Floating point number are essentially scientific notation. Example: 300000000 scientific notation is 3x10^8 .

12)Computers perform scientific notation in what base?
Answer: Computer perform scientific notation with base of 2.

13)What is the problem with adding 1/3 + 1/3 + 1/3 using base 10 and ignoring recurring numbers?
Answer: The conversion of 1/3 with base 10 is 0.3 where 3 is recurring. If human add this number we round up and result is 1.But when computer add this 1/3 with base 10 that is 1/3+ 1/3+1/3 result will be 0.99999999 and then we will run out of digit will stop but it wont round up.

14)What is 1/10 in base 2?
Answer: 1/10 with base 2 is 0.000110011001100110011… we will run out of digit.

15)What is the name of the function discussed in the video?
Answer: The function discussed in this video are Ackermann's function**s.**

16)Can Ackermann's function be coded using for or 'DO' loops?
Answer: Ackermann's function cannot be coded using for or 'DO' loops. Ackermann's function is not a primitive recursive function, i.e. it cannot be defined by loops.

17)What is the value of Ackermann(4,1)?
Answer: Ackermann (4, 1) is 65533.

18)How many minutes will the machine in the video take to calculate Ackermann(4,2)?
Answer: In order to calculate Ackermann (4 ,2), it would actually require lots of time. It would take (2^ 65533 x 3) minutes.

19)The performance characteristic of Ackermann's function is described as what?
Answer: The performance characteristic of Ackermann's function is described as super exponential.

20)A loop nested in another loop has the performance characteristic of what?
Answer: The do loop used in fortan use to go 10 times nested but now a days languages go 256 times nested loop. This kind of looping was necessary to solve the multi dimensional polynomial going on exponetial problems. If two loops are nested then it is sqaured which is multi dimension.

21)What was the limitation of Fortran mentioned in the video?
Answer: The major limitation of FORTRAN mentioned is that it did not do user level recursion. Also, early Fortran compilers didn't allow nested loops to go more than 10 deep.

Name: Gaurav Taneja
ID: 1001955801

22) What real-world use needs complex recursion?
Answer: The real world use for complex recursion but not as difficult as used in ackerman is compilers. Early fortran didn't provide user level recursion but in 1980 they found need for recursion in compilers and use that.

23) There was a need to have a language that could cope with what?
Answer: There need of language that could cope with different width of objects. That there were 16 32 and 8 bits computer.

24) C is most powerful when considered as the classical what?
Answer: It is the most powerful when considered as the classical System Implementation Language**.** The computer's operating system had to be implemented in assembler, which was one of the biggest difficulties in the past. C language was used at this point.

25)What are the names of the two fields of the 'THING' structure?
Answer: The structure is a type definition. There are two fields or member used:
'char *item;': Visually it can be said that it is pointer to the single character but there is possibility it can point to long string. This can be a pointer to the first character of the string.
'struct _thing next;': This is like a recursion now compiler are aware about it performs the computing. This means it calls thing with next that means it point to the part of string.

26) What is the advantage of the 'Triple Ref Technique'?
Answer: Pointers pointing to other pointers are the Triple Ref Pointers. A pointer that points to another pointer, which in turn points to another pointer, is referred to as a triple pointer. The primary benefit of using the Triple Ref approach is that it eliminates the need to maintain track of the location of the previous node. It can be applied when a new node needs to be inserted into the linked list at a certain location. However, we employ a tracer—a Triple Ref Pointer—in the Triple Ref Technique. We can state that a tracer is employed to momentarily point at items. This tracer starts at the head and looks up the value of the following node from the one it is currently on. It determines whether to move to the next node or to insert a new node after the current node by looking at the value of the next node.

27) What is the procedure used in the video to compare the different structures?
Answer: Array is collection of same type of elements. Array the size is decided when the array is created. At the beginning of the linked list it points to nothing but as we add things to the list we allocate memory. The linked list also stores the address of the next thing.
The programs are running on Atari:
I. A linked list and an array are both formed into a struct. The linked list's extra item, which is a pointer to the next item, is the only distinction.
II. Each is given 125,000 random numbers from a generator of random numbers.
III. Next, we add up each element in the linked list. The sum is computed using the same procedure for array. This done 100 times.
IV. Lastly, we keep track of how many clock ticks it takes the function to complete this calculation. To compare the various structures, we examine the times that the list and array take.(linked list takes 166 clock ticks and array takes 179 clock ticks array is slower on atari)

28) Why is the reverse array faster on the Atari?
Answer: The reverse array computation take 114 clock ticks. The code of higher level is converted into the instruction language so backward is favored over forward.

Name: Gaurav Taneja
ID: 1001955801

29) What would be the goal of requiring people to be exposed to coding?
Answer: People who learn to code will have a better understanding of how computers work and will begin to think computationally. If people are introduced to code, they must comprehend how computers operate and how the systems function in order to perform things the way they do. Additionally, it will foster an understanding of how things operate. Young children should be exposed to it so that it opens their minds and they know how to approach a problem. It enables people to think logically.

30) List 3 or more of the different sort algorithms mentioned in the video?
Answer: The following Sorting Algorithms are mentioned in the video:

- Bubble Sort
- Quick Sort
- Selection Sort
- Heap Sort

31)What is the 'Decision Problem'?

Answer: A decision problem is a test to see if a specific formula or function yields a specific result or not. In the decision problem, we're trying to figure out if there's a technique to automatically determine whether or not the supplied premises or assumptions will lead to a conclusion.

32)An example of an abstraction used in the video is, "A transistor is a type of ___"?

Answer: To explain abstraction, it is said that a transistor is a type of switch, that could be turned opened or close. Abstraction is the masking of background information and the presentation of only the relevant data.

33) Which video was the most interesting or your favorite?

Answer: Array vs. Linked List was the video I found to be the most fascinating. On the Atari and Imac, array and linked list processing times were quite flexible. The examination of reverse computing at the end of the video was especially intriguing because the outcomes were incredibly surprising.