Dynamic memory allocation and recursion

Pritha Banerjee

University of Calcutta

cucse.ds@gmail.com

November 5, 2015

Overview

How to use files in C

- Define a file pointer: FILE is a predefined structure defined in stdio.h using typedef. FILE *fp;
- File pointer points to a structure containing informations such as location of buffer, current char position in buffer, whether file is being read/ written, whether end of file is reached etc.
- Need to open a file in read (r), write(w), append(a) mode: fp = fopen("myfile.txt", "r");
- After all your operation, you need to close the file, so that the buffer is released / freed for some other use: fclose(fp)
- To read from a file: fscanf(fp,"%d %s", &ivar, &strvar); To write to file: fprintf(fp, "%d %s", ivar, strvar);
- To check if we have reached end of file: if (feof(fp)) then ...

How to create a single dimension array dynamically?

Size of array depends on the input. How do you generate an array dynamically?

```
void main(){
int  *myarray;
int size;
printf("Enter the size of array : ");
scanf("%d", &size);
myarray = (int *)malloc(sizeof(int) * size);
for (i=0; i<size; i++)
  scanf("%d", &myarray[i]);
free(myarray); }
```

- Allocate 'size' number of integer type memory contiguously.
- myarray points to a location which stores integer type data.
- Free memory after usage using free() function

How to create a multi dimension array dynamically?

Example : How to create 2D array myarray[m][n]?

```
void main(){
int  **myarray;
int size;
printf("Enter the first and second dimension of array : ");
scanf("%d %d", &m, &n);
myarray = (int **)malloc(sizeof(int *) * m);
for (i=0; i<m; i++){
  myarray[i] = (int *)malloc(sizeof(int) * n);
  for (j=0;j<n; j++)
     scanf("%d", &myarray[i][j]);
}
for (i=0; i<m; i++)
  free(myarray[i]);
free(myarray);
}
```

What is recursion?

Recursion

- A function calling itself with certail properties
- There must be a base criteria for which the function does not call itself
- After each call, the function must be closer to base criteria

Example 1: Factorial

$$\begin{aligned} f(n) &= n * f(n-1) && \text{if } n \geq 1 \\ &= 1 && \text{if } n = 0 \end{aligned}$$

Example 2: Fibonacci Sequence

$$\begin{aligned} f(n) &= f(n-1) + f(n-2) && \text{if } n \geq 2 \\ &= 1 && \text{if } 0 \leq n \leq 1 \end{aligned}$$

Recursive code for Fibonacci Sequence

```
unsigned int fib(unsigned int num){
if (num == 0) || (num == 1) return 1;
else return (fib(num - 1) + fib (num-2)); }

int main(void){
 unsigned int m, val;
 printf("Enter which fibonacci sequence m ");
 scanf("%d", &m);
 val = fib(m);
 printf("\nFibonacci ("%d") = %d", m, val); }
```

- Count the number of function calls. Write a program to count the number of function calls.

Coding Rules

1. Name of your source code: dd-mm-yy-<cw? or hw? depending on class/ home work>-<2 digit class roll>. Example: if you are doing Home work 1 given on nov 5, 2015 and your class roll number is 9. your program name should be 05-11-15-hw1-09.c
2. Your code sould include i) problem description in the begining of your file ii) followed by an output of the program iii) followed by time complexity of your algorithm iv) followed by your name and class roll number.
3. Use command -line arguements as much as possible and inputs from files
4. Write functions in a header file that can be used in future.
5. Apply proper indentation, appropriate variable names

Classwork

CW1 Write a program to find if a given integer is present in a list of integers. Use dynamic memory allocation to store list of integers.

CW2 Write a program to transpose a *mxn* matrix and store the result in a different matrix using dynamic memory allocation.

CW3 Write a recursive function *binsearch* to find if an element exists in an array of *n* numbers sorted in increasing order using binary search.

Homework (1): Submission on 19-11-2015

HW1 Write a program to create a 3D array of given dimension p,q,r corresponding to width, height and depth of some items. Generate random integer numbers between 1 to 100 to fill each location of the 3D array.

HW2 Write a program to generate the order of disk replacements in tower of Hanoi recursively.

HW3 Write a recursive procedure to find the minimum of n numbers.

HW4 Write a program to generate all possible permutations of n numbers iteratively

HW5 Write a program to generate all possible permutations of n numbers recursively.

Homework (2): Submission on 19-11-2015

HW6 Write a program *generateinput* to generate random numbers and store them in a file. The format of the command is as follows: generateinput $<$ *number of random number* $>$ $<$ *from* $>$ $<$ *to* $>$ $<$ *filename* $>$. Use command line arguements and file write opertion. Use above program to generate 1000000 random numbers between 0 to 105 and store them in a file.

HW6 Write a program to count the number of people corresponding to each age in the above mentioned population. Use dynamic memory allocation to allocate one dimensional memory. Write a function to show the output in a bar chart

References

 E. Horowitz, S. Sahni and S. A. Freed, Fundamentals of Data Structure in C, 2nd Edition, Universities Press Pvt. Ltd., Chapter 2

The End