Course: CS 211 "Concepts of Algorithms"
Instructor: Dr. Fethi Fkih

Home Work 1

Give the total cost and the complexity (Big O) of the following codes

Code	Cost	Times

`int a = 0, b = 0;`
`for (i = 0; i < N; i++) {`
`  a = a + rand();`
`}`
`for (j = 0; j < M; j++) {`
`  b = b + rand();`
`}`		

Total Cost		
Complexity		

`int a = 0;`
`for (i = 0; i < N; i++) {`
`  for (j = N; j > i; j--) {`
`  a = a + i + j;`
`  }`
`}`		

Total Cost		
Complexity		

`int i, j, k = 0;`
`for (i = n / 2; i <= n; i++) {`
`  for (j = 2; j <= n; j = j * 2)`
`{`
`  k = k + n / 2;`
`  }`
`}`		

Total Cost		
Complexity		

`int a = 0, i = N;`
`while (i > 0) {`
`  a += i;`
`  i /= 2;`		

} **Total Cost** **Complexity**		

for(int i=0;i<n;i++){
  i*=k;
}
Total Cost
Complexity		

int value = 0;
for(int i=0;i<n;i++)
    for(int j=0;j<i;j++)
      value += 1;
Total Cost
Complexity		

int linearSearch(int arr[], int n, int key) {
    for (int i = 0; i < n; i++) {
        if (arr[i] == key) {
            return i;
        }
    }
    return -1;
}
Total Cost
Complexity		

void printPairs(int arr[], int n)}
     for (int i = 0; i < n; i++)}
          for (int j = 0; j < n; j++)}
               printf("(%d, %d)\n", arr[i], arr[j]);
{
{
}
Total Cost
Complexity		

int findMax(int arr[], int n) {
    int max = arr[0]; // Step 1
 	
    for (int i = 1; i < n; i++) {
// Step 2		

`        if (arr[i] > max) { // Step 3`
`            max = arr[i]; // Step 4`
`        }`
`    }`
 	
`    return max; // Step 5`
`}`		

Total Cost		
Complexity		