CS 101 Computer Programming and Utilization

Lecture 12

Recursion

Feb 18, 2011

Prof. R K Joshi
Computer Science and Engineering
IIT Bombay
Email: rkj@cse.iitb.ac.in

Revision: arrays in parameters; recursion

- Int *A A is a pointer
- A pointer variable contains a value which is an address
- We used pointers variables as array names
  - The pointer variable contains the location of beginning of the array
  - Array index can be used
- When array name is passed into a function, a copy of the address is passed
- Using the address inside the formal parameter, one can access original locations and modify them too
  - But one cannot change the array strip outside the function!
    - Due to pass by copy
- Recursion
  - Calling a function from itself
  - Needs termination
  - Simple expressions
  - Eliminate iteration
  - But may be costly at times

Recursion .. a runtime view

```
fact (n) {
  If (n<=1) return 1;
  else return  n * fact (n-1)
}
```

what exactly happens inside the computer when one makes a call such as fact(4) ?

trace of fact (4)

```
fact (n) {
  If (n<=1) return 1;
  else return  n * fact (n-1)
}
main () {… cout << fact (4); }
```

fact (4)

trace of fact (4)

```
fact (n) {
  If (n<=1) return 1;
  else return  n * fact (n-1)
}
main () {… cout << fact (4); }
```

fact (4)

4 * fact (3)

trace of fact (4)

```
fact (n) {
  If (n<=1) return 1;
  else return  n * fact (n-1)
}
main () {… cout << fact (4); }
```

fact (4)

4 * fact (3)

3 * fact (2)

trace of fact (4)

```
fact (n) {
  If (n<=1) return 1;
  else return  n * fact (n-1)
}
main () {… cout << fact (4); }
```

fact (4)

4 * fact (3)

3 * fact (2)

2 * fact (1)

trace of fact (4)

```
fact (n) {
  If (n<=1) return 1;
  else return  n * fact (n-1)
}
main () {… cout << fact (4); }
```

fact (4)

4 * fact (3)

3 * fact (2)

2 * fact (1)

return 1

trace of fact (4)

```
fact (n) {
  If (n<=1) return 1;
  else return  n * fact (n-1)
}
main () {… cout << fact (4); }
```

fact (4)

4 * fact (3)

3 * fact (2)

return 2 * 1

trace of fact (4)

```
fact (n) {
  If (n<=1) return 1;
  else return  n * fact (n-1)
}
main () {… cout << fact (4); }
```

fact (4)

4 * fact (3)

return 3 * 2

trace of fact (4)

```
fact (n) {
  If (n<=1) return 1;
  else return  n * fact (n-1)
}
main () {… cout << fact (4); }
```

fact (4)

return 4 * 6

trace of fact (4)

24

```
fact (n) {
  If (n<=1) return 1;
  else return  n * fact (n-1)
}
main () {… cout << fact (4); }
```

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

fib(1) + fib(0)

1

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

1 + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

1 + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

1 + fib(0)

0

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

1 + 0

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

1 + 0

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

fib(2) + fib(1) fib(1) + fib(0)

1

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

1 + fib(1) fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

1 + fib(1) fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

1 + fib(1) fib(1) + fib(0)

1

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

1 + 1 fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

1 + 1 fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

fib(3) + fib (2)

2 fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

2

fib (4)

\+ fib (2)

fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

2

fib (4)

\+ fib (2)

fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

2

fib (4)

\+ fib (2)

fib(1) + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

2

fib (4)

\+ fib (2)

fib(1) + fib(0)

1

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

2 + fib (2)

1 + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

2 + fib (2)

1 + fib(0)

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

2 + fib (2)

1 + fib(0)

0

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

2 + fib (2)

1 + 0

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

2 + fib (2)

1 + 0

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

2 + fib (2)

1

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

2 + 1

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

2 + 1

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

fib (4)

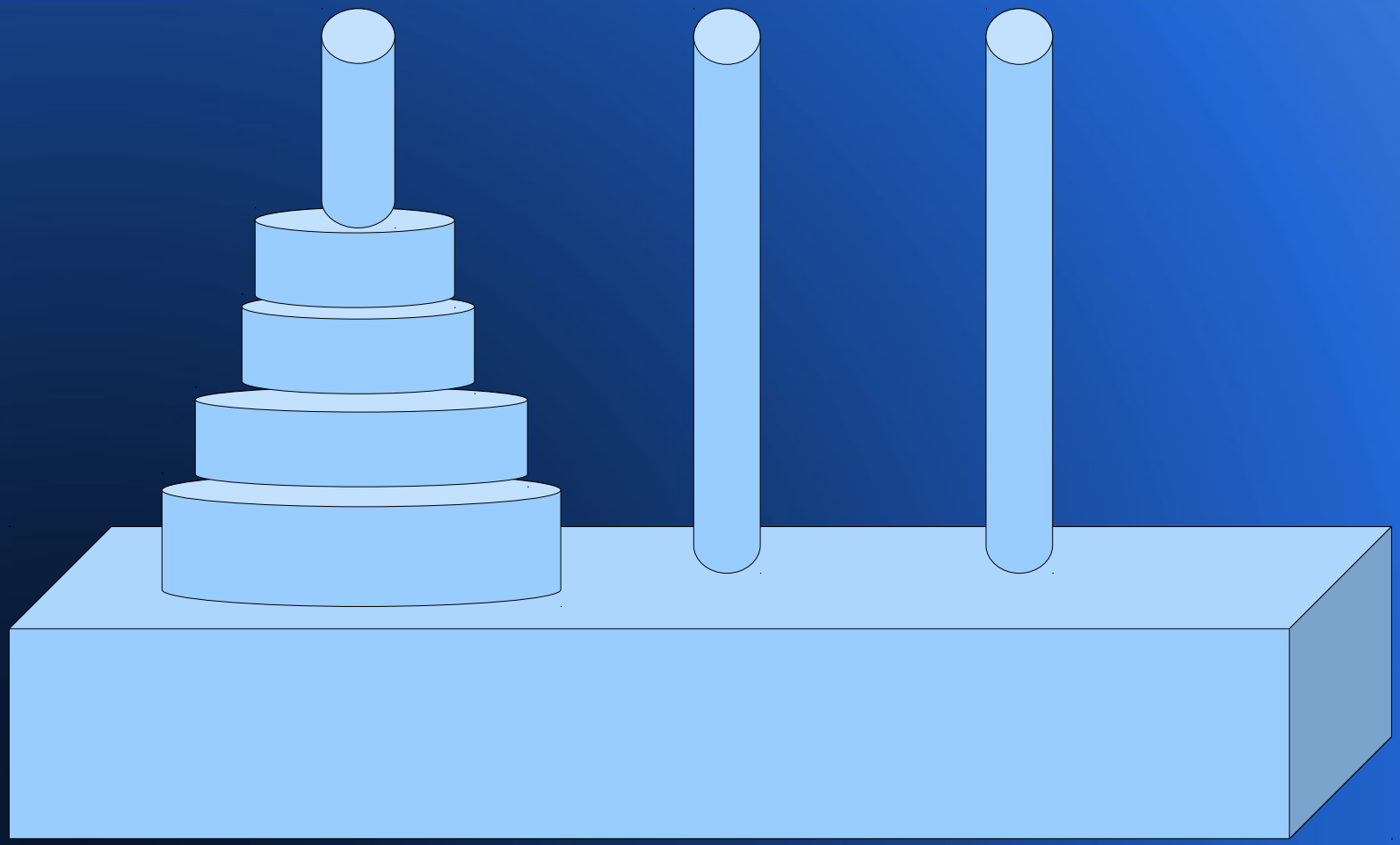
3

Trace of fib(4)

```
fib (n) {
  If (n==1) return 1;
  if (n==0) return 0;
  return  fib(n-1)+fib (n-2)
}
main () {… cout << fib (4); }
```

3

Tower of Hanoi



Euclid's gcd

gcd (14, 21)

gcd (7,14)

gcd (7,7)

gcd (0,7)

7

gcd (30, 90)

gcd (30,60)

gcd (30.30)

gcd (0,30)

30

gcd(7,9)

gcd (2,7)

gcd (2,5)

gcd (2,3)

gcd (1,2)

gcd (1,1)

gcd (0,1)

1