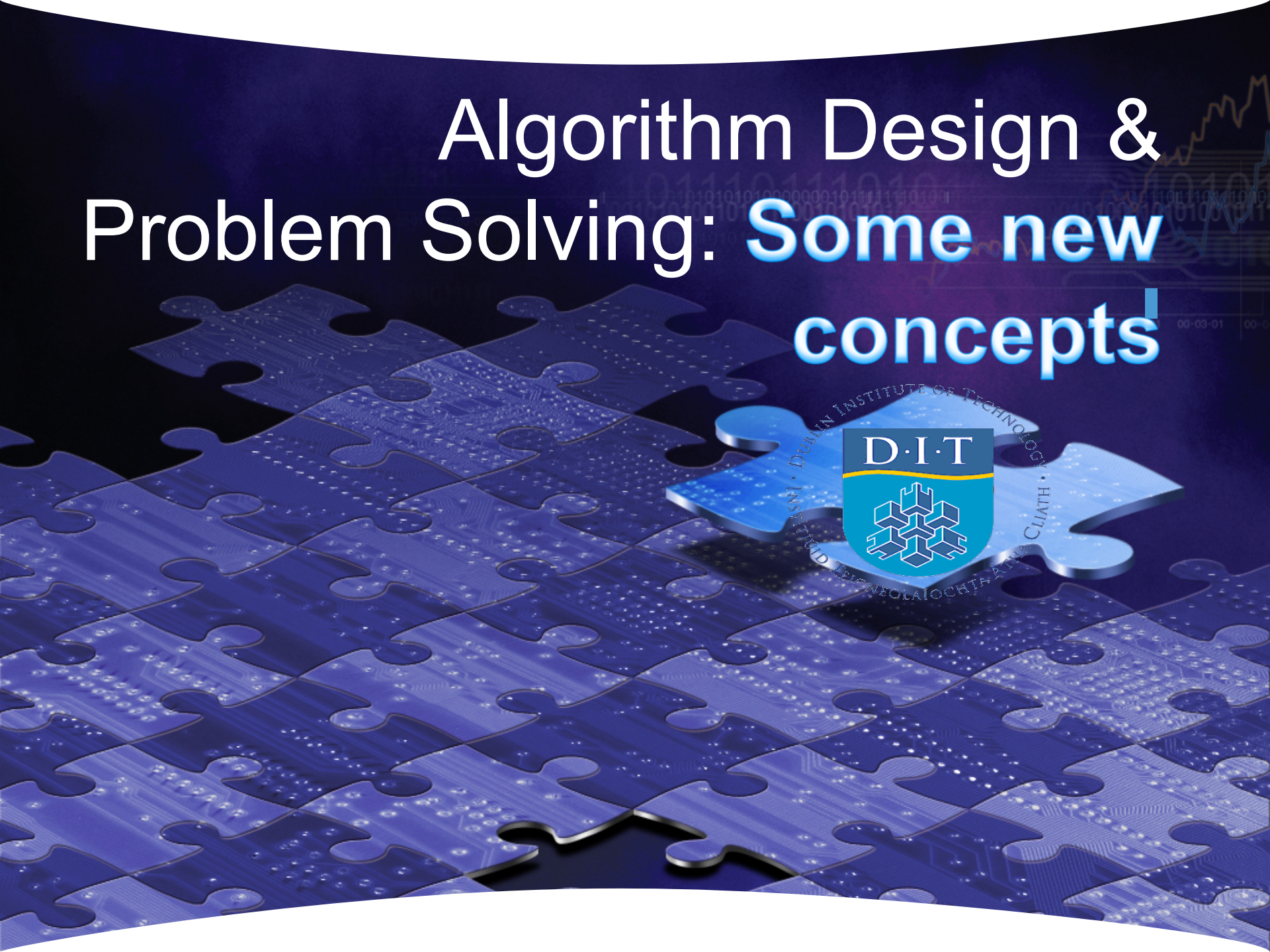
Algorithm Design & Problem Solving: Some new concepts



Contents

Factorial

$$n! = [1*2*3*4* \ldots *n]$$

n! is "n factorial"

Factorial

- Calculate the following:
  - 6!
  - 4!
  - 3!

Factorial

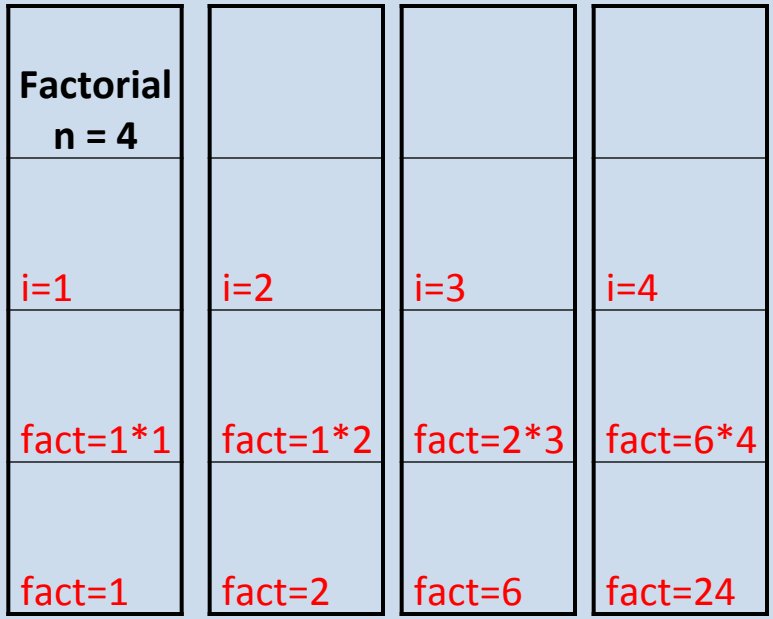
- Write an algorithm to calculate n!

```
Factorial (n)          ------function name
    fact=1             ------variable initialisation
    for i=1 to n       ------loop declaration
        fact=fact*i    ------factorial equation
    return fact        ------send value from function
```

Factorial

- Let's test it with 4! (4*3*2*1=24)

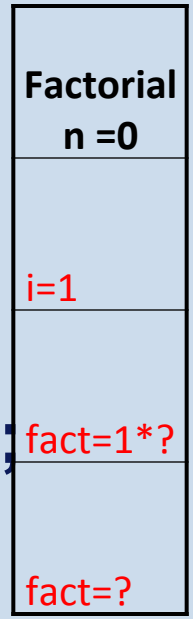
```
Factorial (n)
    fact=1
    for i=1 to n
            fact=fact*i
    return fact
```



Factorial

- Let's test it with 0! (=1)

```
Factorial (n)
    fact=1
    for i=1 to n
            fact=fact*i;
    return fact
```



PROBLEM!
How can we fix it?

Factorial

- An altered algorithm ….

```
Factorial (n)
    fact=1
    if n=1 or n=0
            return 1
    else
    for i=1 to n
            fact=fact*i;
    return fact
```

X^Y

X^Y

- **Calculate the following:**
  - 2 ^ 4
  - 3 ^ 3
  - 4 ^ 2

X^Y

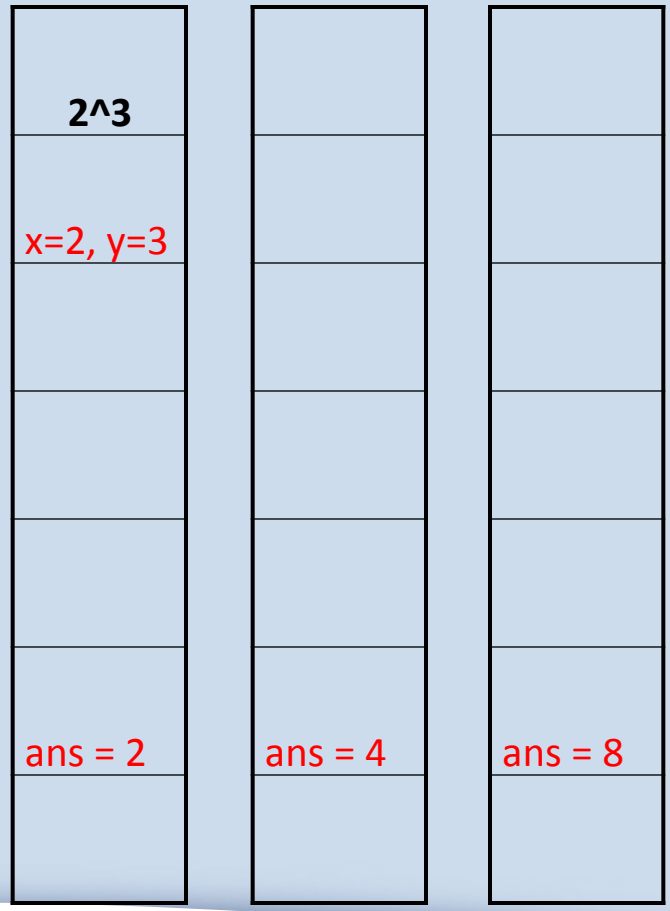
- **Write an algorithm to calculate X ^Y:**

```
Power (x, y)                    ------function name
    ans=1                       ------variable initialisation
    if y=1                      ------if condition
            return 1            ------send value from function
    else
            for i=1 to y        ------loop declaration
                    ans=ans * x; ------power equation
            return ans          ------send value from function
```

X^Y

- Let's test it with 2 ^ 3 (2 * 2 * 2 = 8):

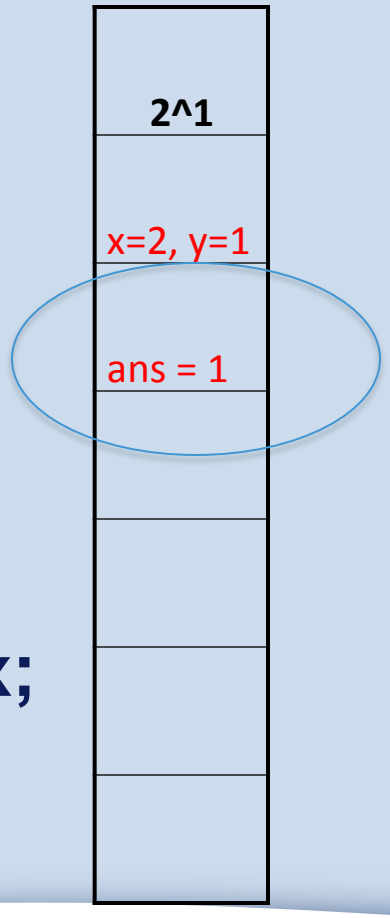
```
Power (x, y)
ans=1
if y=1
    return 1
else
    for i=1 to y
        ans=ans * x;
    return ans
```



X^Y

- Let's test it with 2 ^ 1 (= 2):

```
Power (x, y)
ans=1
if y=1
    return 1
else
    for i=1 to y
            ans=ans * x;
    return ans
```



PROBLEM!
How can we fix it?

X^Y

- An altered algorithm:

```
Power (x, y)
ans=1
if y=0
    return 1
else
    for i=1 to y
            ans=ans * x;
    return ans
```

2 ^ 1 will now be catered for in the for loop

What is Recursion?

When one function calls ITSELF directly or indirectly.

What is Recursion?

- Different mode of thinking.
- Powerful programming tool.
- Divide-and-conquer paradigm.

Recursive Factorial

```
Factorial (n)
    if n=1 or n=0
            return 1
    else
            return n*Factorial(n-1)
```

```
Factorial (n)
fact=1
if n=1 or n=0
    return 1
else
for i=1 to n
            fact=fact*i;
    return fact
```

Recursive X ^ Y

```
Power(x,y)
  if (y=0) then
     return 1;
  else
     return x*Power(x,y-1);
```

```
Power (x, y)
ans=1
if y=0
    return 1
else
    for i=1 to y
            ans=ans * x;
    return ans
```

Calculate GCD

- Given 2 numbers, calculate the greatest common divisor.
- What is “greatest common divisor”?
- It is the largest number that is divisible in a set.
- **What does that mean??**

GCD: An example

(4, 2) The GCD is 2.

Why?

4/2 = 2, 2/2 = 1, **rem=0** Correct

Calculating GCD

- Calculate GCD of the following:

  - **(9, 6)**
  - **(16, 4)**
  - **(20, 16)**

  Now try this …

  - **(72, 32)**

Calculating GCD

How can you calculate GCD if we are using very large numbers?

Use Euclid's Algorithm

Euclid's Algorithm

Find GCD of **(72, 32)**

72, 32 -> 72/32 = 2 rem 8

32, 8 -> 32/8 = 4 rem 0

When rem=0, your divisor is GCD = **8**

Euclid's Algorithm

Find GCD of **(84, 55)**

84, 55 -> 84/55 = 1 rem 29

55, 29 -> 55/29 = 1 rem 26

29, 26 -> 29/26 = 1 rem 3

26, 3 -> 26/3 = 8 rem 2

3, 2 -> 3/2 = 1 rem 1

2, 1 -> 2/1 = 2 rem 0

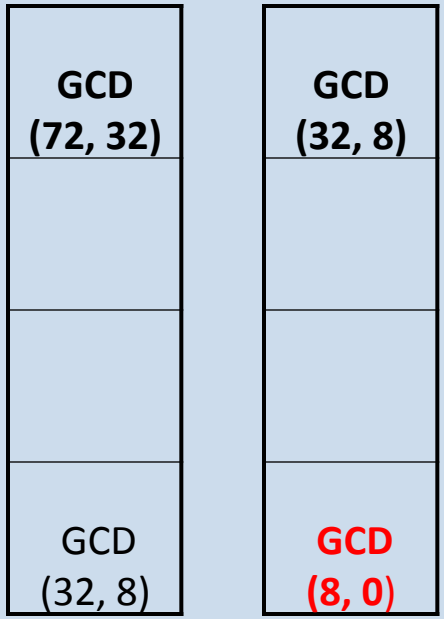
When rem=0, divisor is GCD = **1**

Euclid's Algorithm

Write an algorithm to do this ….

Euclid's Algorithm

```
gcd(a, b)
 if (b = 0) then
    return a
else
    return gcd(b, a mod b)
```



Thank You !


D·I·T
Dublin Institute of Technology
Institiúid Teicneolaíochta Átha Cliath