

Algorithms

COMP 102, lectures 5

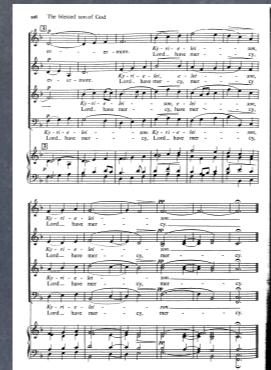
Algorithms

- ⦿ Informal definition

An algorithm is the specification of a sequence of instructions to be carried out by a processor.

Examples

⦿ musical score



⦿ recipe

⦿ assembly instructions



⦿ Origami



⦿ instructions for dialling telephone



⦿ computer program



Music SCORE

116 The blessed son of God

A musical score for three voices (Soprano, Alto, Bass) and piano. The vocal parts are in treble clef, and the piano part is in bass clef. The music is in common time (indicated by '3'). The vocal parts sing in unison, with lyrics in English. The piano part provides harmonic support. The score consists of five systems of music.

System 1: Dynamics: **p**. The vocal parts sing "ev - - - er - more." followed by "Ky - ri - e - lei - - - son," and "Lord - have mer - - - cy," with melodic lines and slurs.

System 2: Dynamics: **p**. The vocal parts sing "ev - er - more." followed by "Ky - ri - e - lei, e - lei - - - son," and "Lord - have mer - cy, mer - - - cy," with melodic lines and slurs.

System 3: Dynamics: **p**. The vocal parts sing "Ky - ri - e - lei - - - son, e - lei - - - son," and "Lord - have mer - - - cy, have mer - - - cy," with melodic lines and slurs.

System 4: Dynamics: **p**. The vocal parts sing "Ky - ri - e - lei - - - son. Ky - ri - e - lei - - - son," and "Lord - have mer - - - cy, Lord - have mer - - - cy," with melodic lines and slurs.

System 5: Dynamics: **p**. The vocal parts sing a final phrase, and the piano part begins its accompaniment. The score ends with dynamics **pp**.

Recipe:

“Gâteau aux Carottes à Johanne”

☞ Instructions

Chauffer le four à 450 F

Beurrer 2 plats de 9 pouces

Dans un grand bol battre le sucre, huile, oeufs et vanille.

Dans un autre bol mélanger farine, soda à pâte, épices et sel

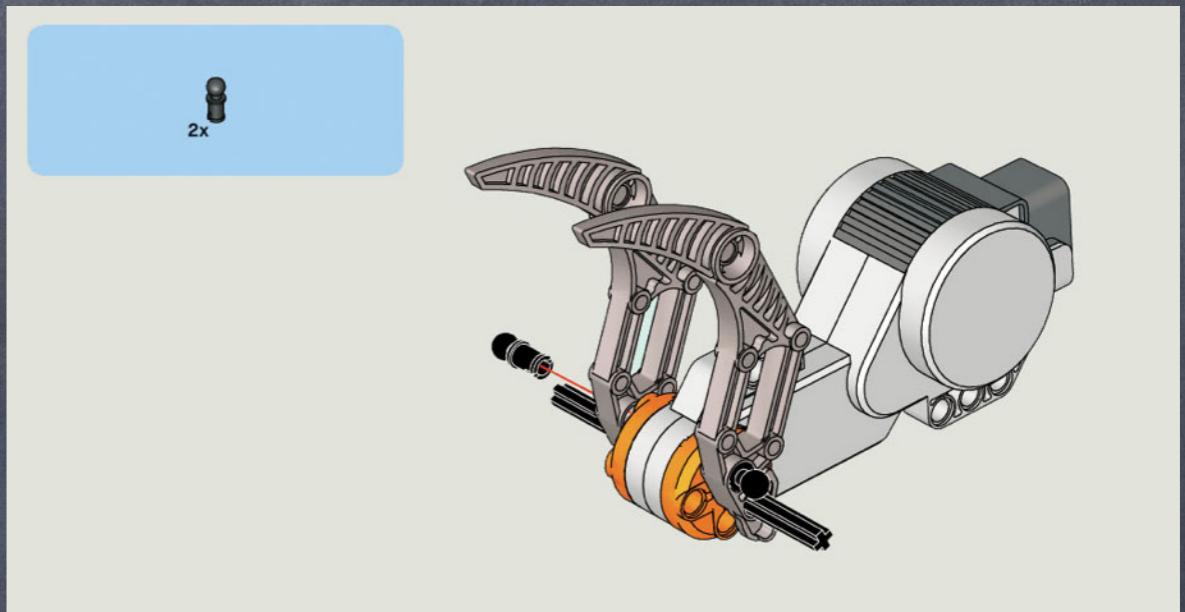
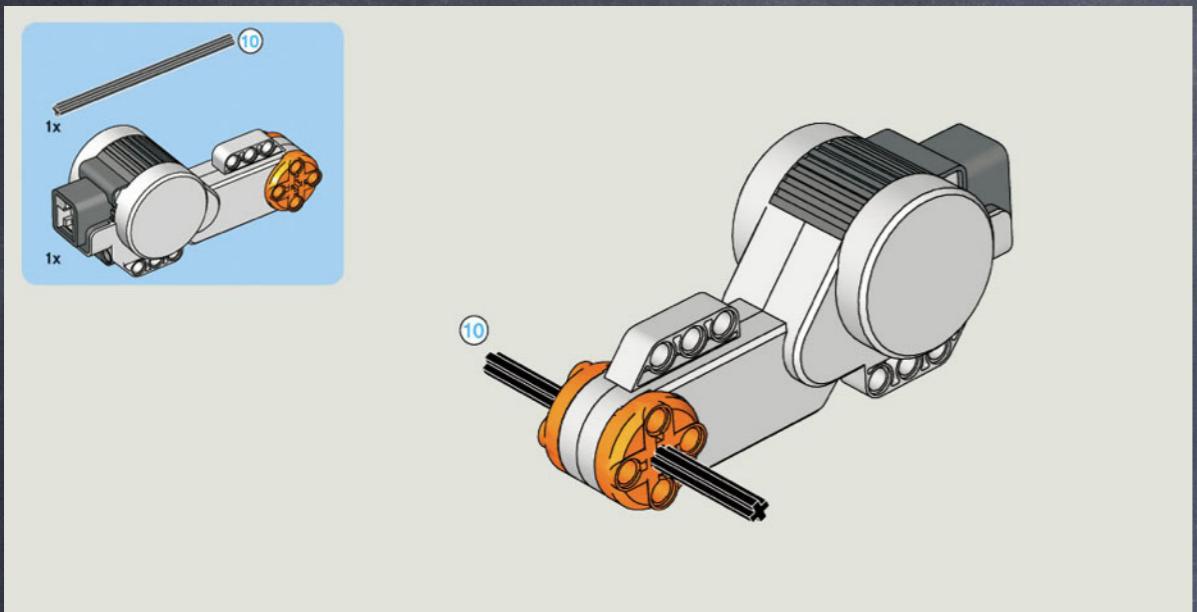
Réunir le tout, ajouter les ananas, carottes et raisins

Placer dans les moules, cuire pendant 40 minutes.

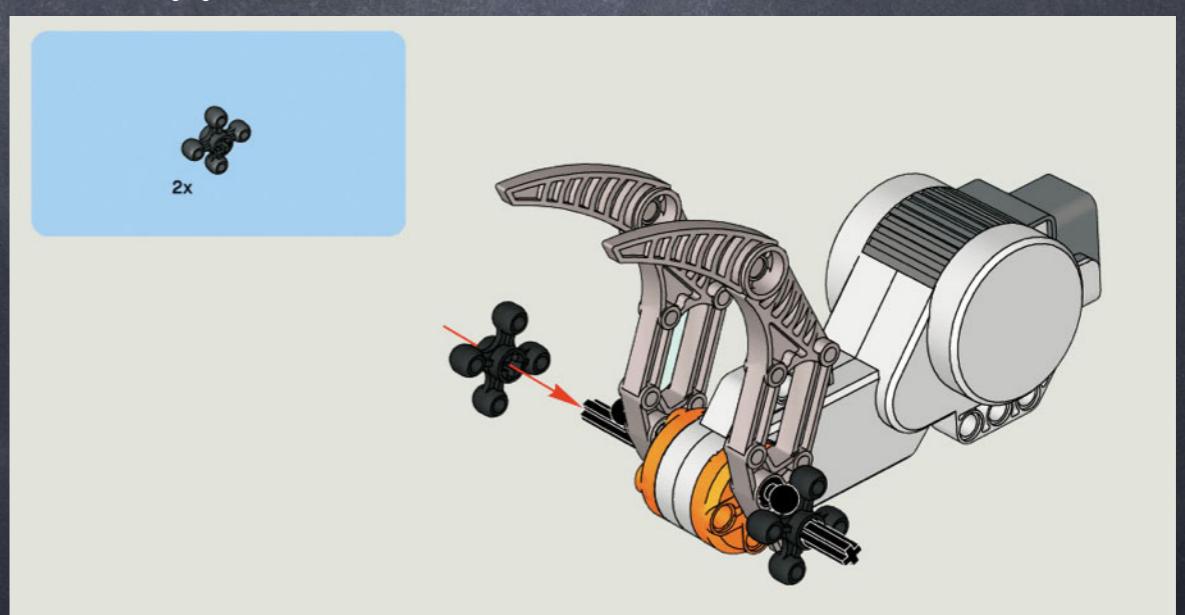
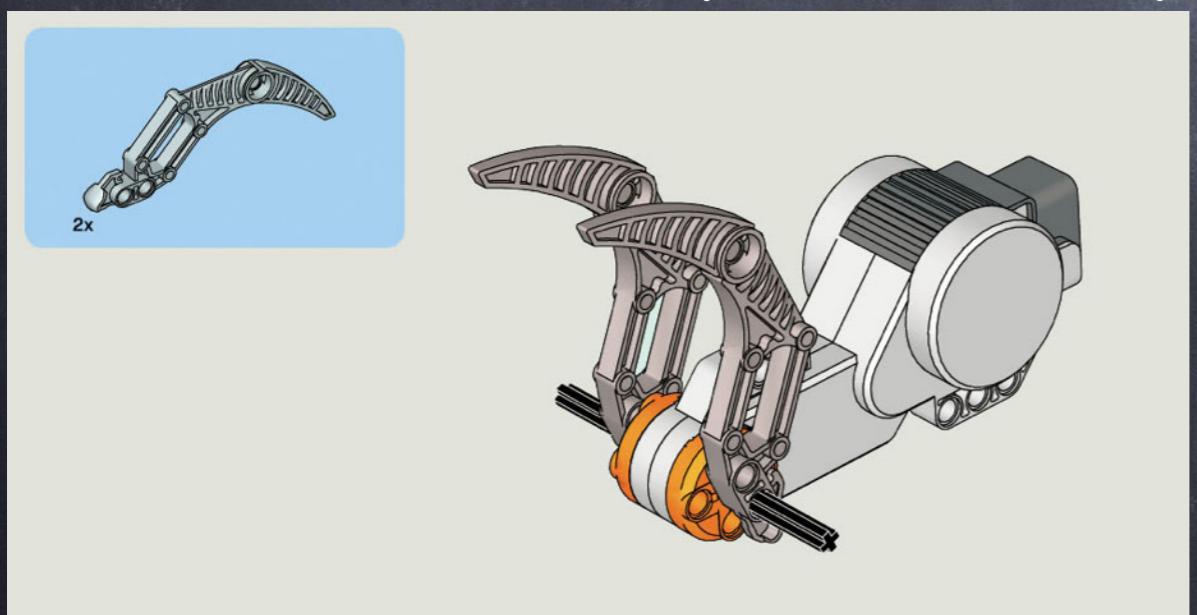
Préparer le glaçage

Laisser refroidir gâteau avant de glacer.

Assembly Instruction

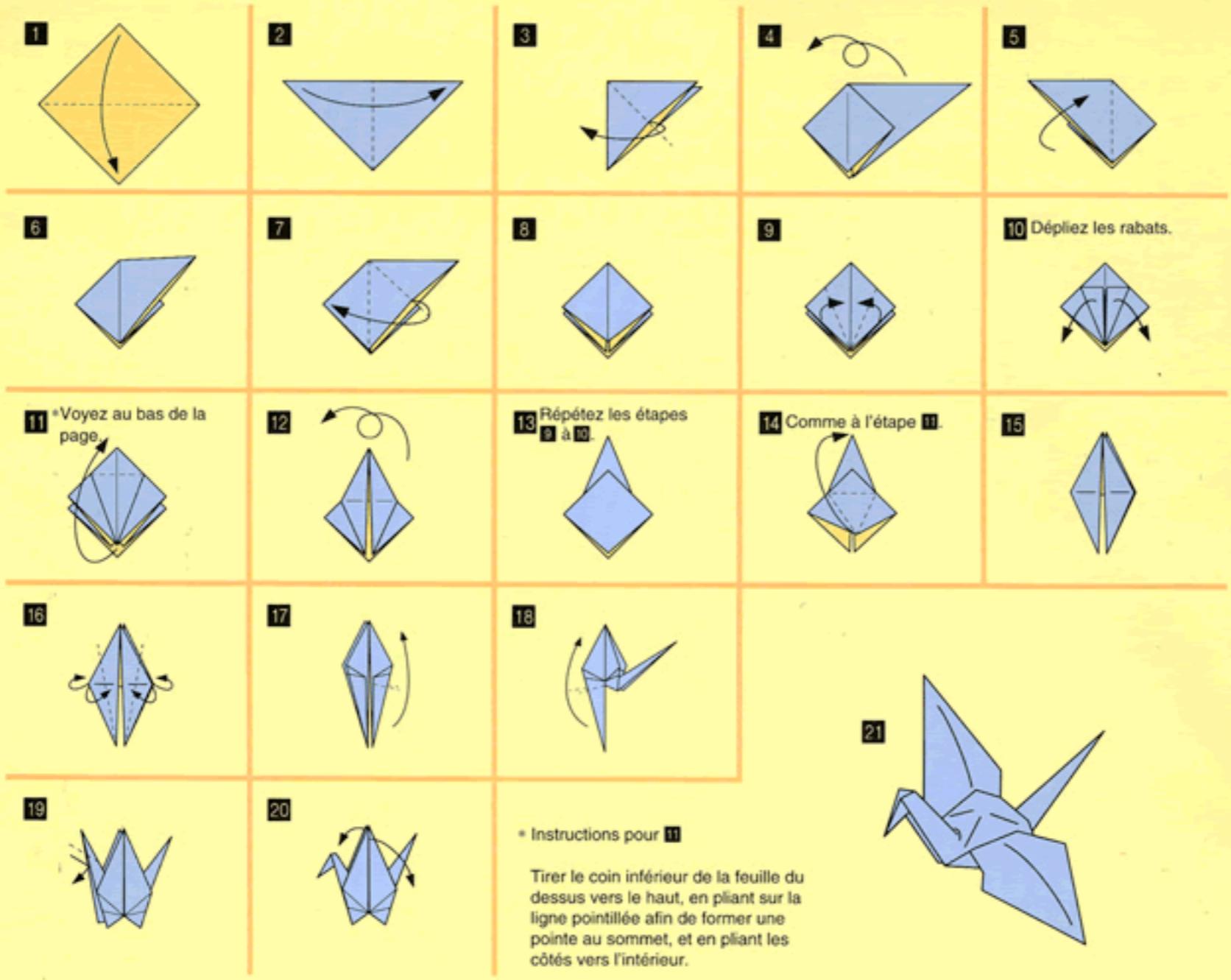


⌚ LEGO (RoboArm (Machine)) instructions



Origami

Comment faire une grue en papier

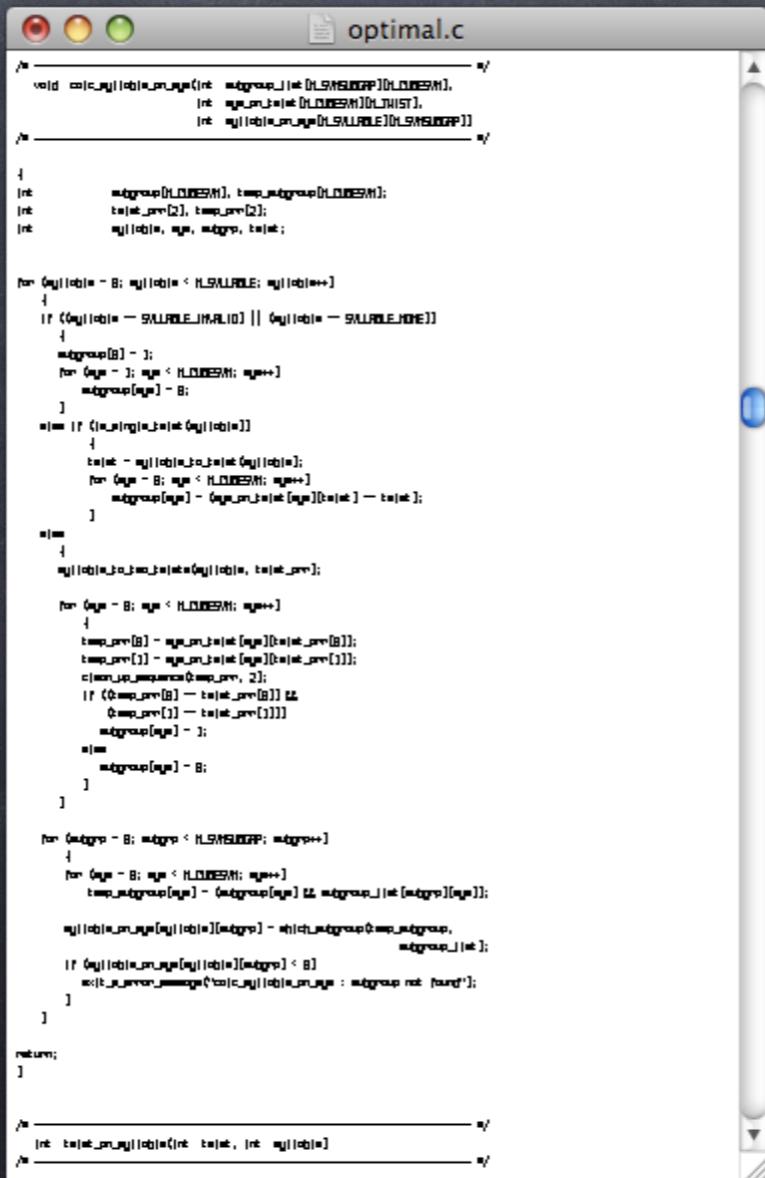


Dialling Instructions



Computer Program

☞ C program



```
#include <cs.h> #include <int.h>
void cs_optimal(int subgroup, int tset[LSHISDGP][LCLDEM], int ap[ap[0][LCLDEM][LCHIST]], int ap[ap[1][LCLDEM][LSWALBE]][LSHISDGP])
{
    int subgroup[LCDESM], temp_subgroup[LCLDEM];
    int ap[ap[2][LCLDEM][LCHIST]];
    int ap[ap[3][LCLDEM][LSWALBE][LCLDEM][LSHISDGP]];

    for (ap[1][ob] = 0; ap[1][ob] < LSWALBE; ap[1][ob]++)
    {
        if ((ap[1][ob] == SWALBE_INVALID) || (ap[1][ob] == SWALBE_NONE))
        {
            subgroup[0] = 0;
            for (ap[2] = 0; ap[2] < LCLDEM; ap[2]++)
                subgroup[ap[2]] = 0;
        }
        else if ((ap[1][ob] <= ap[1][ob] + 1) && (ap[1][ob] >= ap[1][ob] - 1))
        {
            tset[0] = ap[1][ob] * ap[1][ob] * ap[1][ob];
            for (ap[3] = 0; ap[3] < LCLDEM; ap[3]++)
                subgroup[ap[3]] = (ap[3][ob] * ap[3][ob] * ap[3][ob]) - tset[0];
        }
        else
        {
            ap[1][ob] = ap[1][ob] * ap[1][ob] * ap[1][ob];
            for (ap[2] = 0; ap[2] < LCLDEM; ap[2]++)
            {
                temp_subgroup[ap[2]] = ap[2][ob] * ap[2][ob] * ap[2][ob];
                temp_subgroup[1] = ap[2][ob] * ap[2][ob] * ap[2][ob];
                if ((temp_subgroup[0] == tset[0]) &&
                    (temp_subgroup[1] == tset[1]))
                {
                    subgroup[ap[2]] = 1;
                }
                else
                    subgroup[ap[2]] = 0;
            }
        }
    }

    for (subgroup = 0; subgroup < LSHISDGP; subgroup++)
    {
        for (ap[2] = 0; ap[2] < LCLDEM; ap[2]++)
            temp_subgroup[ap[2]] = (subgroup * ap[2]) * ap[2];
        ap[1][ob] = ap[1][ob] * ap[1][ob] * ap[1][ob];
        ap[1][ob] = ap[1][ob] * ap[1][ob] * ap[1][ob];
        if (ap[1][ob] == temp_subgroup[0])
            subgroup[ap[2]] = 1;
        else
            ap[1][ob] = ap[1][ob] * ap[1][ob] * ap[1][ob];
    }
}

int tset_ap(ap[1][ob], ap[1][ob], ap[1][ob])
{
    int ap[ap[0][LCLDEM][LCHIST]], ap[ap[1][LCLDEM][LSWALBE]][LCLDEM][LSHISDGP];
}
```

Computer Science

- ⦿ Computer Science is the study of algorithms for computing machines.
- ⦿ (Formal) Definition of an Algorithm

A **well-ordered** collection of **unambiguous** **effectively computable** operations that when executed produces a **result** and halts in a **finite** amount of time.

What distinguishes computer algorithms ?

- ⦿ Instructions are executed very fast
- ⦿ Little or no human interaction is possible
- ⦿ Algorithm must be fully specified before execution
- ⦿ Algorithm must be unambiguously specified

Example: from decimal representation to integer

```
• input Dn Dn-1 ... D0 (where each Di is 0≤Di≤9)  
val:=0  
pw:=1  
for i:=0 to n do  
    val:=val+pw*Di  
    pw:=pw*10  
output val (val is the integer whose  
decimal representation  
is DnDn-1...D0)
```

D	D	D	D	val	pw	i
5	5	9	1			

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)

val:=0

pw:=1

for i:=0 **to** 3 **do**

i=0: val:=val+pw*D_i ← val:=0+1*D₀ ← val:=0+1*1 ← val:=1

pw:=pw*10 ← pw:=10

i=1: val:=val+pw*D_i ← val:=1+10*D₁ ← val:=1+10*9 ← val:=91

pw:=pw*10 ← pw:=100

i=2: val:=val+pw*D_i ← val:=91+100*D₂ ← val:=91+100*5 ← val:=591

pw:=pw*10 ← pw:=1000

i=3: val:=val+pw*D_i ← val:=591+1000*D₃ ← val:=591+1000*5 ← val:=5591

pw:=pw*10 ← pw:=10000

output val (= 5591)

D	D	D	D	val	pw	i
5	5	9	1			

Example

- **input** 5 5 9 1 (= D₃ D₂ D₁ D₀)

D	D	D	D	val	pw	i
5	5	9	1	0		

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
val:=0

D	D	D	D	val	pw	i
5	5	9	1	0	1	

Example

- **input** 5 5 9 1 (= D₃ D₂ D₁ D₀)
val:=0
pw:=1

D	D	D	D	val	pw	i
5	5	9	1	0	1	0

Example

- **input** 5 5 9 1 (= D₃ D₂ D₁ D₀)
val:=0
pw:=1
for i:=0 **to** 3 **do**

D	D	D	D	val	pw	i
5	5	9	1	0	1	0

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
 val:=0
 pw:=1
for i:=0 **to** 3 **do**
 $i=0: val := val + pw * D_i$

D	D	D	D	val	pw	i
5	5	9	1	0	1	0

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
 val:=0
 pw:=1
for i:=0 **to** 3 **do**
 $i=0: val := val + pw * D_i \leftarrow val := 0 + 1 * D_0$

D	D	D	D	val	pw	i
5	5	9	1	0	1	0

Example

- input 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
 val:=0
 pw:=1
for i:=0 to 3 do
 $i=0: val := val + pw^*D_i \leftarrow val := 0 + 1^*D_0 \leftarrow val := 0 + 1^*1$

D	D	D	D	val	pw	i
5	5	9	1	0	1	0

Example

- input 5 5 9 1 (= $D_3 D_2 D_1 D_0$)

val:=0

pw:=1

for i:=0 to 3 do

i=0: val:=val+pw*D_i ← val:=0+1*D₀ ← val:=0+1*1 ← val:=1

D	D	D	D	val	pw	i
5	5	9	1	1	1	0

Example

- **input** 5 5 9 1 (= D₃ D₂ D₁ D₀)
val:=0
pw:=1
for i:=0 **to** 3 **do**
 i=0: val:=1

D	D	D	D	val	pw	i
5	5	9	1	1	1	0

Example

- **input** 5 5 9 1 (= D₃ D₂ D₁ D₀)
 val:=0
 pw:=1
for i:=0 **to** 3 **do**
 i=0: val:=1
 pw:=pw*10

D	D	D	D	val	pw	i
5	5	9	1	1	1	0

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
 val:=0
 pw:=1
for i:=0 **to** 3 **do**
 i=0: val:=1
 pw:=pw*10 ← pw:=10

D	D	D	D	val	pw	i
5	5	9	1	1	10	0

Example

- **input** 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 **to** 3 **do**

i=0: val:=1

pw:=10

D	D	D	D	val	pw	i
5	5	9	1	1	10	1

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

i=0: val:=1

pw:=10

i=1: val:=val+pw*D_i

D	D	D	D	val	pw	i
5	5	9	1	1	10	1

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
- val:=0
- pw:=1
- for** i:=0 **to** 3 **do**
- i=0: val:=1
- pw:=10
- i=1: val:=val+pw*D_i ← val:=1+10*D₁

D	D	D	D	val	pw	i
5	5	9	1	1	10	1

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

i=0: val:=1

pw:=10

i=1: val:=val+pw*D_i ← val:=1+10*D₁ ← val:=1+10*9

D	D	D	D	val	pw	i
5	5	9	1	1	10	1

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

i=0: val:=1

pw:=10

i=1: val:=val+pw*D_i ← val:=1+10*D₁ ← val:=1+10*9 ← val:=91

D	D	D	D	val	pw	i
5	5	9	1	91	10	1

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

 i=0: val:=1

 pw:=10

 i=1: val:=91

D	D	D	D	val	pw	i
5	5	9	1	91	10	1

Example

- **input** 5 5 9 1 (= D₃ D₂ D₁ D₀)
- val:=0
- pw:=1
- for** i:=0 **to** 3 **do**
- i=0: val:=1
- pw:=10
- i=1: val:=91
- pw:=pw*10

D	D	D	D	val	pw	i
5	5	9	1	91	10	1

Example

- **input** 5 5 9 1 (= D₃ D₂ D₁ D₀)
- val:=0
- pw:=1
- for** i:=0 **to** 3 **do**
- i=0: val:=1
- pw:=10
- i=1: val:=91
- pw:=pw*10 ← pw:=100

D	D	D	D	val	pw	i
5	5	9	1	91	100	1

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

 i=0: val:=1

 pw:=10

 i=1: val:=91

 pw:=100

D	D	D	D	val	pw	i
5	5	9	1	91	100	2

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

i=0: val:=1

pw:=10

i=1: val:=91

pw:=100

i=2: val:=val+pw*D_i

D	D	D	D	val	pw	i
5	5	9	1	91	100	2

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
- val:=0
- pw:=1
- for** i:=0 **to** 3 **do**
- i=0: val:=1
- pw:=10
- i=1: val:=91
- pw:=100
- i=2: val:=val+pw*D_i ← val:=91+100*D₂

D	D	D	D	val	pw	i
5	5	9	1	91	100	2

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
- val:=0
- pw:=1
- for** i:=0 **to** 3 **do**
- i=0: val:=1
- pw:=10
- i=1: val:=91
- pw:=100
- i=2: val:=val+pw*D_i ← val:=91+100*D₂ ← val:=91+100*5

D	D	D	D	val	pw	i
5	5	9	1	91	100	2

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
- val:=0
- pw:=1
- for** i:=0 **to** 3 **do**
- i=0: val:=1
- pw:=10
- i=1: val:=91
- pw:=100
- i=2: val:=val+pw*D_i ← val:=91+100*D₂ ← val:=91+100*5 ← val:=591

D	D	D	D	val	pw	i
5	5	9	1	591	100	2

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

 i=0: val:=1

 pw:=10

 i=1: val:=91

 pw:=100

 i=2: val:=591

D	D	D	D	val	pw	i
5	5	9	1	591	100	2

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

 i=0: val:=1

 pw:=10

 i=1: val:=91

 pw:=100

 i=2: val:=591

 pw:=pw*10

D	D	D	D	val	pw	i
5	5	9	1	591	100	2

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

i=0: val:=1

pw:=10

i=1: val:=91

pw:=100

i=2: val:=591

pw:=pw*10 ← pw:=1000

D	D	D	D	val	pw	i
5	5	9	1	591	1000	2

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

 i=0: val:=1

 pw:=10

 i=1: val:=91

 pw:=100

 i=2: val:=591

 pw:=1000

D	D	D	D	val	pw	i
5	5	9	1	591	1000	3

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

i=0: val:=1

pw:=10

i=1: val:=91

pw:=100

i=2: val:=591

pw:=1000

i=3: val:=val+pw*D_i

D	D	D	D	val	pw	i
5	5	9	1	591	1000	3

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
- val:=0
- pw:=1
- for** i:=0 **to** 3 **do**
- i=0: val:=1
- pw:=10
- i=1: val:=91
- pw:=100
- i=2: val:=591
- pw:=1000
- i=3: val:=val+pw*D_i ← val:=591+1000*D₃

D	D	D	D	val	pw	i
5	5	9	1	591	1000	3

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

i=0: val:=1

pw:=10

i=1: val:=91

pw:=100

i=2: val:=591

pw:=1000

i=3: val:=val+pw*D_i ← val:=591+1000*D₃ ← val:=591+1000*5

D	D	D	D	val	pw	i
5	5	9	1	591	1000	3

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

i=0: val:=1

pw:=10

i=1: val:=91

pw:=100

i=2: val:=591

pw:=1000

i=3: val:=val+pw*D_i ← val:=591+1000*D₃ ← val:=591+1000*5 ← val:=5591

D	D	D	D	val	pw	i
5	5	9	1	5591	1000	3

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

 i=0: val:=1

 pw:=10

 i=1: val:=91

 pw:=100

 i=2: val:=591

 pw:=1000

 i=3: val:=5591

D	D	D	D	val	pw	i
5	5	9	1	5591	1000	3

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
- val:=0
- pw:=1
- for** i:=0 **to** 3 **do**
- i=0: val:=1
- pw:=10
- i=1: val:=91
- pw:=100
- i=2: val:=591
- pw:=1000
- i=3: val:=5591
- pw:=pw*10

D	D	D	D	val	pw	i
5	5	9	1	5591	1000	3

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

i=0: val:=1

pw:=10

i=1: val:=91

pw:=100

i=2: val:=591

pw:=1000

i=3: val:=5591

pw:=pw*10 ← pw:=10000

D	D	D	D	val	pw	i
5	5	9	1	5591	10000	3

Example

- input 5 5 9 1 (= D₃ D₂ D₁ D₀)

val:=0

pw:=1

for i:=0 to 3 do

 i=0: val:=1

 pw:=10

 i=1: val:=91

 pw:=100

 i=2: val:=591

 pw:=1000

 i=3: val:=5591

 pw:=10000

D	D	D	D	val	pw	i
5	5	9	1	5591	10000	3

Example

- **input** 5 5 9 1 (= $D_3 D_2 D_1 D_0$)
- val:=0
- pw:=1
- for** i:=0 **to** 3 **do**
- i=0: val:=1
- pw:=10
- i=1: val:=91
- pw:=100
- i=2: val:=591
- pw:=1000
- i=3: val:=5591
- pw:=10000
- output** val (= 5591)