# Algorithm Improvement Course

Onnes

2022/12/10

## Contents

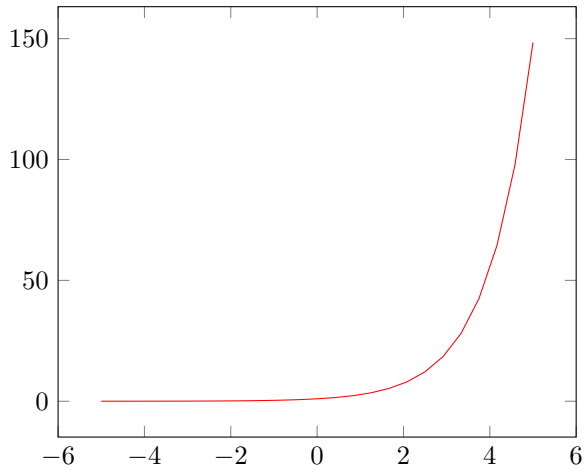
# 1 1.dynamic programming

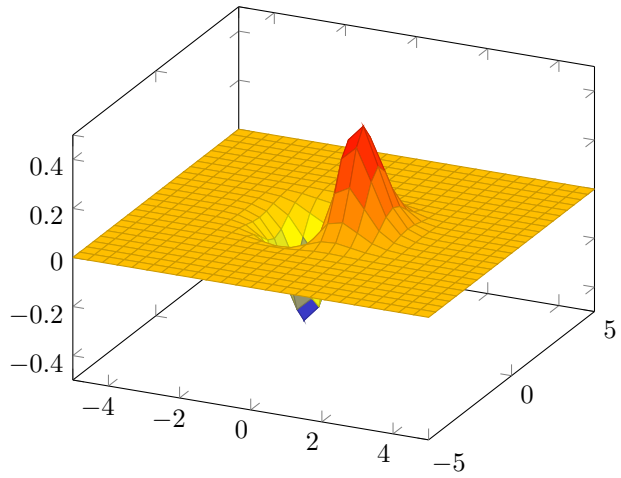
## 1.1 1.digital triangle model

### 1.1.1 Acwing 1015.picking peanuts

**question** Acwing 1015.picking peanuts
time limit per test : 1 second
memory limit per test: 64 megabytes
input:standard input
output:standard output

Hello Kitty wants to pick some peanuts for her favorite Mickey Mouse.
She came to a rectangular peanut field with grid-like roads (pictured below), entering from the northwest corner and exiting from the southeast corner.
At the intersection of each road in the field, there is a peanut seedling with several peanuts on it, and all the peanuts on it can be picked off after passing through a peanut seedling.
Hello Kitty can only go east or south, not west or north.
Ask Hello Kitty how many peanuts can be picked at most.




input

> The first line is an integer T, representing how many sets of data there are.
> Next is the T group of data.
> The first line of each set of data is two integers, representing the number of rows R and the number of columns C of peanut seedlings.
> The next R row of data in each group of data describes the situation of each row of peanut seedlings in sequence from north to south. There are C integers in each row of data, and the number M of peanuts on each peanut seedling in the row is described in order from west to east.

output

> For each set of input data, output a line containing the maximum number of peanuts that Hello Kitty can pick.

range

> $1 \leq T \leq 100$
> $1 \leq R,\ C \leq 100$
> $0 \leq M \leq 100$

input

```
2
2 2
1 1
3 4
2 3
2 3 4
1 6 5
```

output

```
8
16
```

**idea**

**Theorem 1**

$x^2 + y^2 = r^2$

```cpp
#include <bits/stdc++.h>

using namespace std;

const int N = 1e5 + 10;
int a[N];

int main() {
    int n; // input
    scanf("%d", &n);
    puts("Hello world!");
    for (int i = 0; i < n; i ++ ) {
        printf("%d", a[i]);
    }
    return 0;
}
```