Q1.

T(n)=n5.

Let f(n)=2n, C=1, N=23.

Then for all n≥N, T(n)≤C\*2n → T(n)=O(2n)

Q2.

T(n)=100n+logn.

Let f(n)=n+(logn)2, C=100,N取对数函数的底数.

Then for all n≥N，T(n) ≤ 100\*(n+logn)<100\*(n+(logn)2) → T(n)=O(n+(logn)2).

Q3.

T(n) , k≥d,

Let f(n)=O(nk), C=max{a1…ai…ad}, N=1.

Then for all n≥N, T(n) ≤ =C\*d\*nk → T(n)=O(nk)

Q4.

T(n) =(n+a)b

Let f(n)=nb,

Q5.

1. 遍历所有的数

Numbers of 0：0

Numbers of 1：4

Numbers of 2：3

Numbers of 3：1

Numbers of 4：2

Numbers of 5：1

Numbers of 6：0

Numbers of 7：1

Numbers of 8：0

Numbers of 9：1

1. 输出

1，1，1，1，2，2，2，3，4，4，5，7，9

Q6.

**Algorithm**:

A[n];B[m];i=j=0;

**While** i<n **and** j<m **do**

**if** A[i]==B[j] then j++, i++;

**else** i++;

**end**

**return** j==m;

复杂度分析：B序列是A序列的子序列，从低位依次比较序列元素，如果元素相等则比较各自序列下一个元素；如果不等，子序列保持不变，主序列移到下一个元素。最坏情况需要遍历整个主序列的元素，所以时间复杂度为O(n).

Q7.

a=10, b=3, d=2.5, f(n)= n2.5

因为a\*f(n/b)=10\*(n/3)2.5=10/32.5\*n2.5,记作k\*f(n),易得k<1，所以T(n)= Θ(f(n))= Θ(n2.5)

Q8.

由题意可得：F(n)=2F(n/2)+1,其中a=2, b=2, f(n)=1

因为a\*f(n/b)=2\*f(n), K=2>1, 所以d=logba=1, F(n)= Θ(nd)= Θ(n)

Q9.

**Algorithm**: 最长公共子序列伪代码

A[n];B[m];

dp[n+1][m+1]=0;

**for** i=1 to n **do**

**for** j=1 to m **do**

**if** A[i]==B[j] **then** dp[i][j]=dp[i-1][j-1]+1;

**else** dp[i][j]=Max(dp[i][j-1], dp[i-1][j])

**end**

**end**

**return** dp[n][m];

Q10.

**Algorithm**: 序列对齐伪代码

A[n];B[m];

P[n+1][m+1]=0;σ;α;

**for** i=1 to n do

**for** j=1 to m do

**if** A[i]==B[j] **then** min=P[i-1][j-1];

**else** min= P[i-1][j-1]+ α;

min=Min(min, σ+P[i-1][j], σ+P[i][j-1])

**end**

**end**

**return** P[n][m];

Q11.

Step1:choose s; update: s→C:2; s→A:8;

Step2:choose C; update:s→A:5; s→B:3;

Step3:choose B; update:s→A:4; s→t:9;

Step4:choose A; update:s→E:5; s→D:5; s→t:9;

Step5:choose D; update:s→E:5; s→t:7;

Step6:choose E; update:s→t:7;

Step7:choose t;

Path of s→t: s→C→B→A→D→t; Length of this path: 7

Q12.

Q13.