

# A

我们对  $d$  作十进制分解:  $d = \sum_{i=0}^n p_i \times 10^i$ , 其中  $0 < p_i < 10$  且为整数, 可得  $f(d) = \sum_{i=0}^n p_i$ , 可以 **逐项比较** 得出两者大小, 此处不再赘述。

并且, 我们可以得到这样一个 **推论**:

- $d$  的增长比  $f(d)$  来得快。

可以感性理解:  $d$  增长得过程中会遇到逢十进一得情况, 这样对  $d$  来说是好的, 可以  $f(d)$  的几个 9 就会变成 1, 是亏损的。

总之, 我们只需要找出第一个  $RBN$ , 就可以肯定大于等于这个数的所有数都满足要求。

```
// using ll = long long
void solve() {
    string ss;
    cin >> n >> m;
    ll l = 1, r = n, ans = n + 1, mid;
    while (l <= r) {
        mid = l + r >> 1;
        ll tt = mid;
        ll sum = 0;
        while (tt) {
            sum += tt % 10;
            tt /= 10;
        }
        if (mid - sum >= m) {
            r = mid - 1;
            ans = mid;
        } else {
            l = mid + 1;
        }
    }
    cout << n - ans + 1;
    cout << endl;
}
```

# F

题意要求, 至少存在一条长度为4的路径能从起点出发经过其他点一次并返回, 考虑 *bfs* 枚举起点的四个方向的四次*bfs*, 只要存在一个点, 他的下一步可以到起点并且此时距离 $dis_i$ 大于等于3, 那么这条路线合法。

```
void solve()
{
    string ss;
    cin >> n >> m;
    vector<string> a(n+1);
    ll sx, sy;
    vector<vector<bool>> vis(n+2, vector<bool> (m+2, 1));
    rep(i, 1, n+1){
        cin >> a[i];
    }
```

```

a[i] = '#' + a[i];
repi(j, 1, m + 1) {
    if (a[i][j] == 'S')
        sx = i, sy = j;
    if (a[i][j] == '.') {
        vis[i][j] = 0;
    } else {
        vis[i][j] = 1;
    }
}
}
queue<Pa> q;
auto check = [&](ll x, ll y) -> bool {
    if (x > n || x < 1) return 0;
    if (y > m || y < 1) return 0;
    if (a[x][y] == '#') return 0;
    return 1;
};
bool ok = 0;
rep(j, 4) {
    ll st = sx + mx[j];
    ll sr = sy + my[j];
    if (!vis[st][sr]) {
        q.push({st, sr});
        vis[sx][sy] = 1;
        vis[st][sr] = 1;
        while (!q.empty()) {
            Pa tt = q.front();
            q.pop();
            FOR(4) {
                x = tt.first + mx[i];
                y = tt.second + my[i];
                if (check(x, y)) {
                    if (!vis[x][y]) {
                        vis[x][y] = 1;
                        q.push({x, y});
                    }
                }
            }
        }
    } else if (check(st, sr)) {
        YES;
        return;
    }
}
}
NO;
}

```

