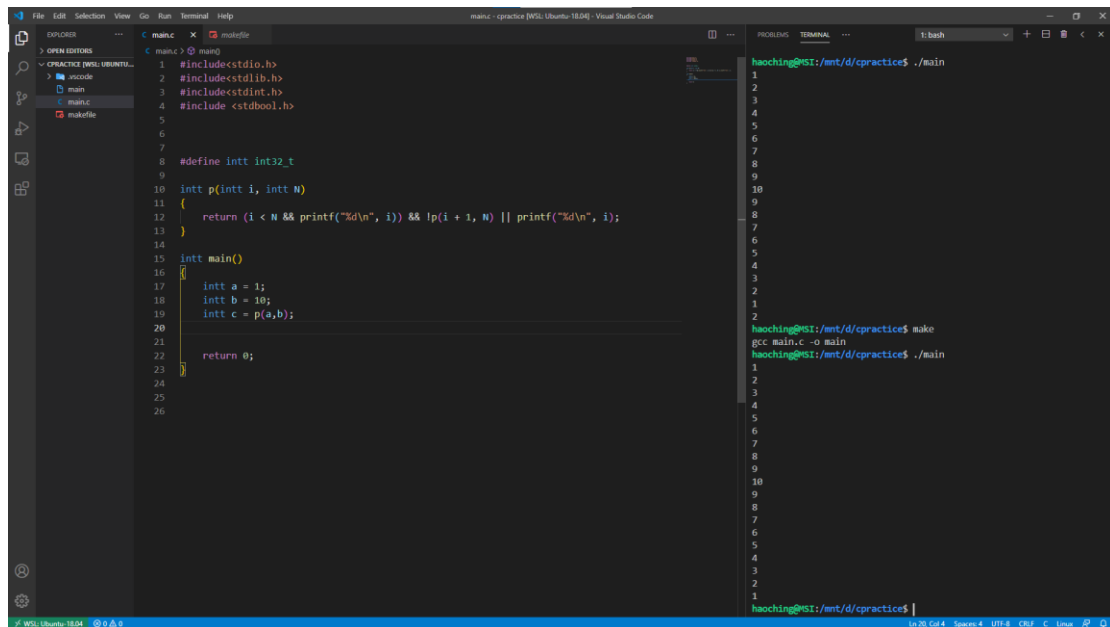


Bonus: Code Explanation

當 $i < N$ 時會執行 `&&` 之後的動作 輸出 i ，接下來進入到函式 `p(i+1,N)`當中接著輸出 $i + 1, i + 2, \dots, N - 1$ ，當 $i = N$ 時 $i < N$ 不會成立則跳至 `||` 後方輸出 $i (N)$ 出來，接著回到上一函式輸出 $N - 1, N - 2, \dots, i$ 並結束函式。



The screenshot shows a Visual Studio Code editor with a C program in `main.c` and a terminal window showing the execution results. The C program defines a recursive function `p` that prints numbers from i to $N-1$ and then calls itself with $i+1$. The `main` function sets `a=1`, `b=10`, and calls `p(a,b)`. The terminal output shows the sequence of numbers printed during the execution of `p(1,10)`.

```
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<stdint.h>
4 #include <stdbool.h>
5
6
7
8 #define intt int32_t
9
10 intt p(intt i, intt N)
11 {
12     return (i < N && printf("%d\n", i) && !p(i + 1, N) || printf("%d\n", i);
13 }
14
15 intt main()
16 {
17     intt a = 1;
18     intt b = 10;
19     intt c = p(a,b);
20
21
22     return 0;
23 }
24
25
26
```

```
haoching@MSI:/mnt/d/cpractice$ ./main
1
2
3
4
5
6
7
8
9
10
haoching@MSI:/mnt/d/cpractice$ make
gcc main.c -o main
haoching@MSI:/mnt/d/cpractice$ ./main
1
2
3
4
5
6
7
8
9
10
9
8
7
6
5
4
3
2
1
haoching@MSI:/mnt/d/cpractice$
```

當 $i = 1, N = 10$ 時的輸出結果