Algorithm assignment #2

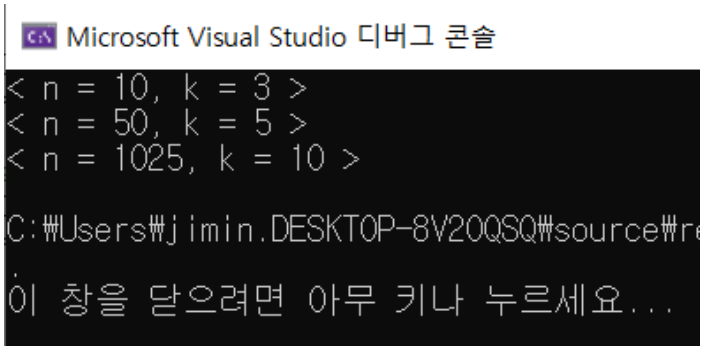
응용통계학과 20202850 김지민

P1.

- **code**

```
#include <stdio.h>

int power(int x, int n) {
	int y = 0;
	if (n == 0) {
		return 1;
	}
	if (n % 2 == 1) { // n is odd
		y = power(x, (n - 1) / 2);
		return x * y * y;
	}
	else { // n is even
		y = power(x, n / 2);
		return y * y;
	}
}
int main() {
	int arr[3] = { 10, 50, 1025 };
	int k;
	for (int i = 0; i < 3; i++) {
		k = 0;
		while (power(2, k) <= arr[i]) {
			k++;
		}
		printf("< n = %d, k = %d >₩n", arr[i], k - 1);
	}

}
```

- **result**

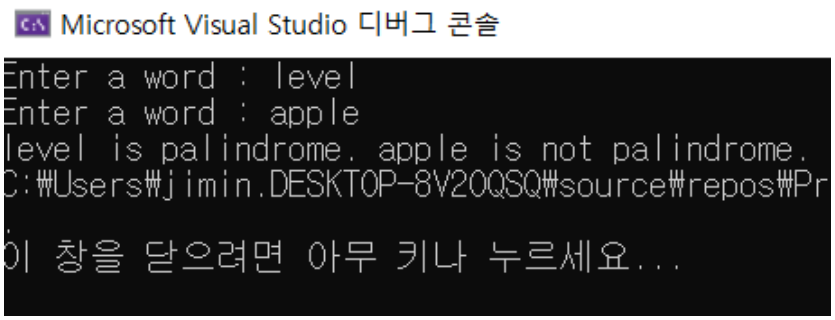


P2.

- **code**

```c
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <stdbool.h>
#include <stdlib.h>
#include <string.h>
#define MAX_NUM 100

void isPalindrome(char* word) {
    int len = strlen(word);

    if (len <= 1) {
        fprintf(stderr, "%s is not allowed. Please enter at least two characters.", word);
        exit(1);
    }

    bool isPalindrome = true;
    for (int i = 0; i < len / 2; i++) { // Repeat from 0 to half the length of the string
        if (word[i] != word[len - 1 - i]) {
            isPalindrome = false;
            break;
        }
    }

    if (isPalindrome == true) {
        printf("%s is palindrome. ", word);
    }
    else {
        printf("%s is not palindrome. ", word);
    }
}

int main()
{
    char word[MAX_NUM], word2[MAX_NUM];

    printf("Enter a word : ");
    scanf("%s", word);

    printf("Enter a word : ");
    scanf("%s", word2);

    isPalindrome(word);
    isPalindrome(word2);

    return 0;
}
```

- **result**



P3.

int x=3, y=2, z=5;

printf("%d₩n", ((x>y)?x:y)>z?((y>x)?x:y):z);

1) (x > y) ? x : y

   (3 > 2) ? 3 : 2

   Since (x > y) is true, the one of the left is the result. So, output is 3.

2) (y > x) ? x : y

   (2 > 3) ? 3 : 2

   Since (y > x) is false, the one of the right is the result. So, output is 2.

3) (3 > z) ? 2 : z

   (3 > 5) ? 2 : 5

   Since (3 > z) is false, the one of the right is the result. So, output is 5.

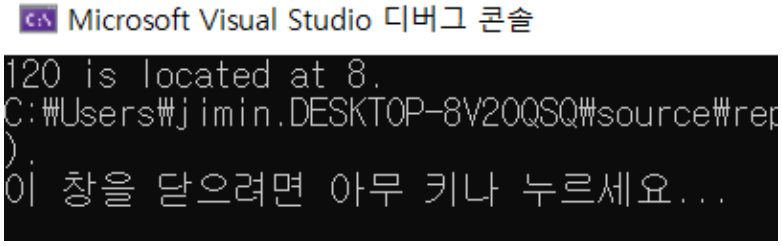
Therefore, the output of the following code is 5.

P4.

- **code**

```
#include <stdio.h>
#include <stdbool.h>

void binary_search(int* num, int n, int key) {
	int low = 0;
	int high = n-1;
	int mid;
	bool keyinlist = false;

	while (low <= high) {

		mid = (low + high) / 2;

		if (num[mid] > key) {
			high = mid - 1;
		}
		else if (num[mid] < key) {
			low = mid + 1;
		}
		else {
			printf("%d is located at %d.", key, mid+1);
			keyinlist = true;
			break;
		}
	}
	if (!keyinlist) {
		printf("The key is not in the list.");
	}
}

int main() {
	int num[9] = { 12, 34, 37, 45, 57, 82, 99, 120, 134 };
	int key = 120;

	binary_search(num, 9, key);
}
```

- **result**

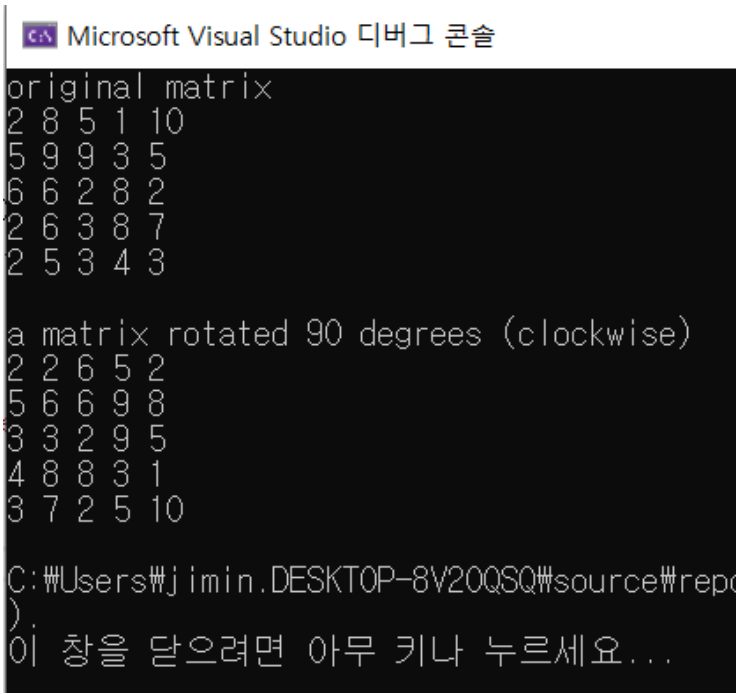


P5.

- **code**

```
#include <stdio.h>
#include <stdlib.h>

int main() {
	int matrix[5][5];
	printf("original matrix₩n");
	for (int i = 0; i < 5; i++) {
		for (int j = 0; j < 5; j++) {
			matrix[i][j] = (rand() % 10) + 1;
			printf("%d ", matrix[i][j]);
		}
		printf("₩n");
	}

	printf("₩na matrix rotated 90 degrees (clockwise)₩n");
	int temp[5][5];
	for (int i = 0; i < 5; i++) {
		for (int j = 0; j < 5; j++) {
			temp[i][j] = matrix[4 - j][i];
			printf("%d ", temp[i][j]);
		}
		printf("₩n");
	}
}
```

- **result**

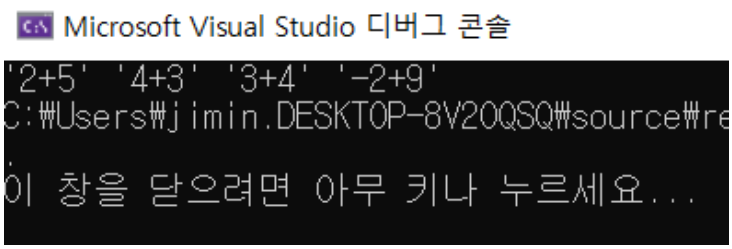


P6.

- **code**

```
#include <stdio.h>

void pairSum(int arr[], int num, int length) {
    for (int i = 0; i < length; i++) {
        for (int j = i + 1; j < length; j++) {
            if (arr[i] + arr[j] == num) {
                printf("'%d+%d' ", arr[i], arr[j]);
            }
        }
    }
}

int main() {
    int arr[10] = { 2,4,3,5,6,-2,4,7,8,9 };
    int length = sizeof(arr) / sizeof(int);
    pairSum(arr, 7, length);

}
```

- **result**



P7.

- **code**

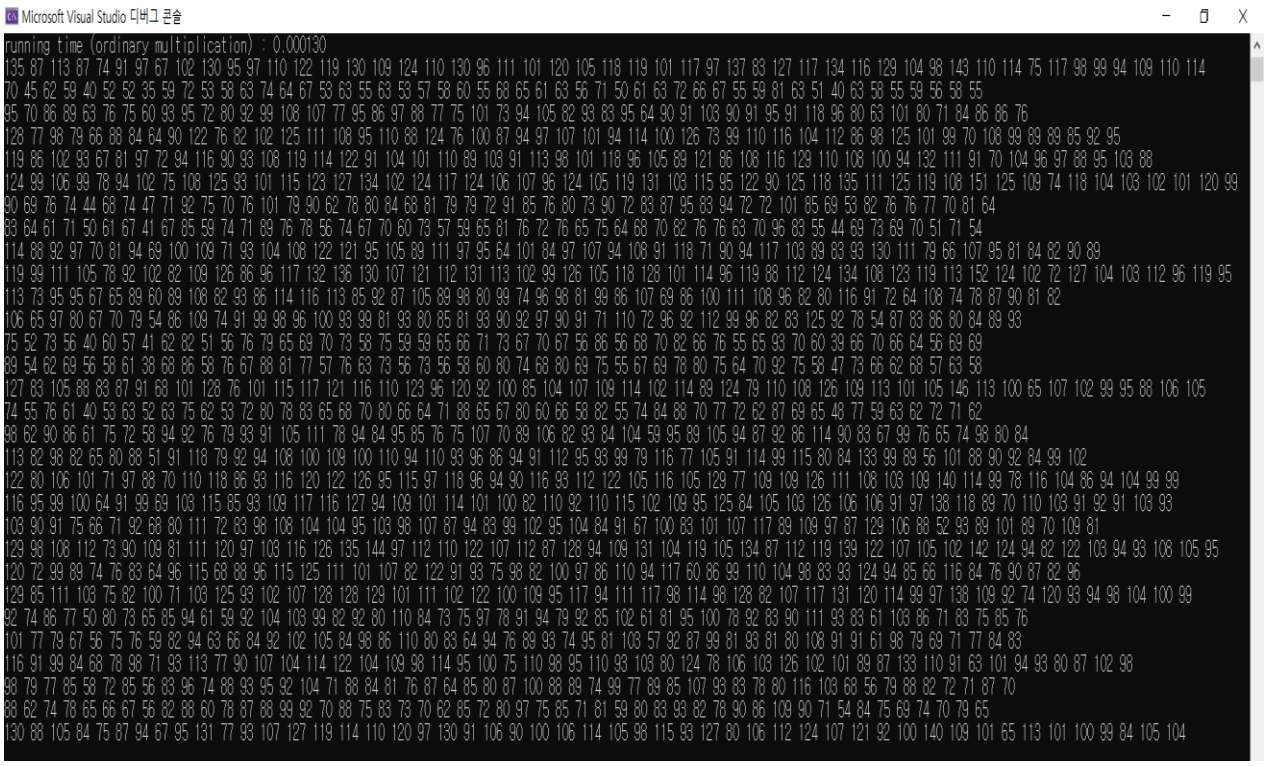
```
#include <stdio.h>
#include <stdlib.h>
#include <windows.h>

int main(void) {

	int matrix[30][10];
	for (int i = 0; i < 30; i++) {
		for (int j = 0; j < 10; j++) {
			matrix[i][j] = (rand() % 5) + 1;
		}
	}

	int matrix2[10][50];
	for (int i = 0; i < 10; i++) {
		for (int j = 0; j < 50; j++) {
			matrix2[i][j] = (rand() % 5) + 1;
		}
	}

	LARGE_INTEGER Frequency, start_ticks, end_ticks, diff_ticks;

	QueryPerformanceFrequency(&Frequency);
	QueryPerformanceCounter(&start_ticks);

	int result1[30][50] = { 0, };

	for (int i = 0; i < 30; i++) {
		for (int j = 0; j < 50; j++) {
			for (int k = 0; k < 10; k++) {
				result1[i][j] += matrix[i][k] * matrix2[k][j];
			}
		}
	}

	QueryPerformanceCounter(&end_ticks);

	diff_ticks.QuadPart = end_ticks.QuadPart - start_ticks.QuadPart;
	double duringtime = (double)diff_ticks.QuadPart / (double)Frequency.QuadPart;
	printf("running time (ordinary multiplication) : %f₩n", duringtime);

	for (int i = 0; i < 30; i++) {
		for (int j = 0; j < 50; j++) {
			printf("%d ", result1[i][j]);
		}
		printf("₩n");
	}

	LARGE_INTEGER Frequency2, start_ticks2, end_ticks2, diff_ticks2;

	QueryPerformanceFrequency(&Frequency2);
```

```
QueryPerformanceCounter(&start_ticks2);

int a[15][5], b[15][5], c[15][5], d[15][5];
for (int i = 0; i < 15; i++) { // split matrix
        for (int j = 0; j < 5; j++) {
                a[i][j] = matrix[i][j];
                b[i][j] = matrix[i][j + 5];
                c[i][j] = matrix[i + 15][j];
                d[i][j] = matrix[i + 15][j + 5];
        }
}

int e[5][25], f[5][25], g[5][25], h[5][25];
for (int i = 0; i < 5; i++) { // split matrix2
        for (int j = 0; j < 25; j++) {
                e[i][j] = matrix2[i][j];
                f[i][j] = matrix2[i][j + 25];
                g[i][j] = matrix2[i + 5][j];
                h[i][j] = matrix2[i + 5][j + 25];
        }
}

int f_h[5][25], g_e[5][25], e_h[5][25], g_h[5][25], e_f[5][25];
for (int i = 0; i < 5; i++) {
        for (int j = 0; j < 25; j++) {
                f_h[i][j] = f[i][j] - h[i][j]; // f-h
                g_e[i][j] = g[i][j] - e[i][j]; // g-e
                e_h[i][j] = e[i][j] + h[i][j]; // e+h
                g_h[i][j] = g[i][j] + h[i][j]; // g+h
                e_f[i][j] = e[i][j] + f[i][j]; // e+f
        }
}

int a_b[15][5], c_d[15][5], a_d[15][5], b_d[15][5], a_c[15][5];
for (int i = 0; i < 15; i++) {
        for (int j = 0; j < 5; j++) {
                a_b[i][j] = a[i][j] + b[i][j]; // a+b
                c_d[i][j] = c[i][j] + d[i][j]; // c+d
                a_d[i][j] = a[i][j] + d[i][j]; // a+d
                b_d[i][j] = b[i][j] - d[i][j]; // b+d
                a_c[i][j] = a[i][j] - c[i][j]; // a-c
        }
}

int p1[15][25] = { 0, };
int p2[15][25] = { 0, };
int p3[15][25] = { 0, };
int p4[15][25] = { 0, };
int p5[15][25] = { 0, };
int p6[15][25] = { 0, };
int p7[15][25] = { 0, };

for (int i = 0; i < 15; i++) {
        for (int j = 0; j < 25; j++) {
                for (int k = 0; k < 5; k++) {
```

```
                                p1[i][j] += a[i][k] * f_h[k][j]; // p1 = a*(f-h)
                                p2[i][j] += a_b[i][k] * h[k][j]; // p2 = (a+b)*h
                                p3[i][j] += c_d[i][k] * e[k][j]; // p3 = (c+d)*e
                                p4[i][j] += d[i][k] * g_e[k][j]; // p4 = d*(g-e)
                                p5[i][j] += a_d[i][k] * e_h[k][j]; // p5 = (a+d)*(e+h)
                                p6[i][j] += b_d[i][k] * g_h[k][j]; // p6 = (b-d)*(g+h)
                                p7[i][j] += a_c[i][k] * e_f[k][j]; // p7 = (a-c)*(e+f)
                        }
                }
        }

        int r[15][25];
        int s[15][25];
        int t[15][25];
        int u[15][25];

        for (int i = 0; i < 15; i++) {
                for (int j = 0; j < 25; j++) {
                        r[i][j] = p5[i][j] + p4[i][j] - p2[i][j] + p6[i][j]; // r = p5+p4-
p2+p6
                        s[i][j] = p1[i][j] + p2[i][j]; // s = p1+p2
                        t[i][j] = p3[i][j] + p4[i][j]; // t = p3+p4
                        u[i][j] = p5[i][j] + p1[i][j] - p3[i][j] - p7[i][j]; // u=p5+p1-p3-p7
                }
        }

        int result2[30][50];
        for (int i = 0; i < 15; i++) { // merge matrix
                for (int j = 0; j < 25; j++) {
                        result2[i][j] = r[i][j];
                        result2[i][j + 25] = s[i][j];
                        result2[i + 15][j] = t[i][j];
                        result2[i + 15][j + 25] = u[i][j];
                }
        }

        QueryPerformanceCounter(&end_ticks2);

        diff_ticks2.QuadPart = end_ticks2.QuadPart - start_ticks2.QuadPart;
        double duringtime2 = (double)diff_ticks2.QuadPart / (double)Frequency2.QuadPart;
        printf("₩nrunning time (using the Strassen's idea) : %f₩n", duringtime2);

        for (int i = 0; i < 30; i++) {
                for (int j = 0; j < 50; j++) {
                        printf("%d ", result2[i][j]);
                }
                printf("₩n");
        }

        return 0;
}
```

- **result (same result)**

Running time (ordinary multiplication) : 0.000130(ms)



Running time (using the Strassen's idea) : 0.000083(ms)

```
running time (using the Strassen's idea) : 0.000083
135 87 113 87 74 91 97 67 102 130 95 97 110 122 119 130 109 124 110 130 96 111 101 120 105 118 119 101 117 97 137 83 127 117 134 116 129 104 98 143 110 114 75 117 98 99 94 109 110 114
70 45 62 59 40 52 52 35 59 72 53 58 63 74 64 67 53 63 55 63 53 57 58 60 55 68 65 61 63 56 71 50 61 63 72 66 67 55 59 81 63 51 40 63 58 55 59 56 58 55
95 70 86 89 63 76 75 60 93 95 72 80 92 99 108 107 77 95 86 97 88 77 75 101 73 94 105 82 93 83 95 64 90 91 103 90 91 95 91 118 96 80 63 101 80 71 84 86 86 76
128 77 98 79 66 88 84 64 90 122 76 82 102 125 111 108 95 110 88 124 76 100 87 94 97 107 101 94 114 100 126 73 99 110 116 104 112 86 98 125 101 99 70 108 99 89 89 85 92 95
119 86 102 93 67 81 97 72 94 116 90 93 108 119 114 122 91 104 101 110 89 103 91 113 98 101 118 96 105 89 121 86 108 116 129 110 108 100 94 132 111 91 70 104 96 97 88 95 103 88
124 99 106 99 78 94 102 75 108 125 93 101 115 123 127 134 102 124 117 124 106 107 96 124 105 119 131 103 115 95 122 90 125 118 135 111 125 119 108 151 125 109 74 118 104 103 102 101 120 99
90 69 76 74 44 68 74 47 71 92 75 70 76 101 79 90 62 78 80 84 68 81 79 79 72 91 85 76 80 73 90 72 83 87 95 83 94 72 72 101 85 69 53 82 76 76 77 70 81 64
83 64 61 71 50 61 67 41 67 85 59 74 71 89 76 78 56 74 67 70 60 73 57 59 65 81 76 72 76 65 75 64 68 70 82 76 76 63 70 96 83 55 44 69 73 69 70 51 71 54
114 88 92 97 70 81 94 69 100 109 71 93 104 108 122 121 95 105 89 111 97 95 64 101 84 97 107 94 108 91 118 71 90 94 117 103 89 83 93 130 111 79 66 107 95 81 84 82 90 89
119 99 111 105 78 92 102 82 109 126 86 96 117 132 136 130 107 121 112 131 113 102 99 126 105 118 128 101 114 96 119 88 112 124 134 108 123 119 113 152 124 102 72 127 104 103 112 96 119 95
113 73 95 95 67 65 89 60 89 108 82 93 86 114 116 113 85 92 87 105 89 98 80 99 74 96 98 81 99 86 107 69 86 100 111 108 96 82 80 116 91 72 64 108 74 78 87 90 81 82
106 65 97 80 67 70 79 54 86 109 74 91 99 98 96 100 93 99 81 93 80 85 81 93 90 92 97 90 91 71 110 72 96 92 112 99 96 82 83 125 92 78 54 87 83 86 80 84 89 93
75 52 73 56 40 60 57 41 62 82 51 56 76 79 65 69 70 73 58 75 59 59 65 66 71 73 67 70 67 56 86 56 68 70 82 66 76 55 65 93 70 60 39 66 70 66 64 56 69 69
89 54 62 69 56 58 61 38 68 86 58 76 67 88 81 77 57 76 63 73 56 73 56 58 60 80 74 68 80 69 75 55 67 69 78 80 75 64 70 92 75 58 47 73 66 62 68 57 63 58
127 83 105 88 83 87 91 68 101 128 76 101 115 117 121 116 110 123 96 120 92 100 85 104 107 109 114 102 114 89 124 79 110 108 126 109 113 101 105 146 113 100 65 107 102 99 95 88 106 105
74 55 76 61 40 53 63 52 63 75 62 53 72 80 78 83 65 68 70 80 66 64 71 88 65 67 80 60 66 58 82 55 74 84 88 70 77 72 62 87 69 65 48 77 59 63 62 72 71 62
98 62 90 86 61 75 72 58 94 92 76 79 93 91 105 111 78 94 84 95 85 76 75 107 70 89 106 82 93 84 104 59 95 89 105 94 87 92 86 114 90 83 67 99 76 65 74 98 80 84
113 82 98 82 65 80 88 51 91 118 79 92 94 108 100 109 100 110 94 110 93 96 86 94 91 112 95 93 99 79 116 77 105 91 114 99 115 80 84 133 99 89 56 101 88 90 92 84 99 102
122 80 106 101 71 97 88 70 110 118 86 93 116 120 122 126 95 115 97 118 96 94 90 116 93 112 122 105 116 105 129 77 109 109 126 111 108 103 109 140 114 99 78 116 104 86 94 104 99 99
116 95 99 100 64 91 99 69 103 115 85 93 109 117 116 127 94 109 101 114 101 100 82 110 92 110 115 102 109 95 125 84 105 103 126 106 106 91 97 138 118 89 70 110 103 91 92 91 103 93
103 90 91 75 66 71 92 68 80 111 72 83 98 108 104 104 95 103 98 107 87 94 83 99 102 95 104 84 91 67 100 83 101 107 117 89 109 97 87 129 106 88 52 93 89 101 89 70 109 81
129 98 108 112 73 90 109 81 111 120 97 103 116 126 135 144 97 112 110 122 107 112 87 128 94 109 131 104 119 105 134 87 112 119 139 122 107 105 102 142 124 94 82 122 103 94 93 108 105 95
120 72 99 89 74 76 83 64 96 115 68 88 96 115 125 111 101 107 82 122 91 93 75 98 82 100 97 86 110 94 117 60 86 99 110 104 98 83 93 124 94 85 66 116 84 76 90 87 82 96
129 85 111 103 75 82 100 71 103 125 93 102 107 128 128 129 101 111 102 122 100 109 95 117 94 111 117 98 114 98 128 82 107 117 131 120 114 99 97 138 109 92 74 120 93 94 98 104 100 99
92 74 86 77 50 80 73 65 85 94 61 59 92 104 103 99 82 92 80 110 84 73 75 97 78 91 94 79 92 85 102 61 81 95 100 78 92 83 90 111 93 83 61 103 86 71 83 75 85 76
101 77 79 67 56 75 76 59 82 94 63 66 84 92 102 105 84 98 86 110 80 83 64 94 76 89 93 74 95 81 103 57 92 87 99 81 93 81 80 108 91 91 61 98 79 69 71 77 84 83
116 91 99 84 68 78 98 71 93 113 77 90 107 104 114 122 104 109 98 114 95 100 75 110 98 95 110 93 103 80 124 78 106 103 126 102 101 89 87 133 110 91 63 101 94 93 80 87 102 98
98 79 77 85 58 72 85 56 83 96 74 88 93 95 92 104 71 88 84 81 76 87 64 85 80 87 100 88 89 74 99 77 89 85 107 93 83 78 80 116 103 68 56 79 88 82 72 71 87 70
88 62 74 78 65 66 67 56 82 88 60 78 87 88 99 92 70 88 75 83 73 70 62 85 72 80 97 75 85 71 81 59 80 83 93 82 78 90 86 109 90 71 54 84 75 69 74 70 79 65
130 88 105 84 75 87 94 67 95 131 77 93 107 127 119 114 110 120 97 130 91 106 90 100 106 114 105 98 115 93 127 80 106 112 124 107 121 92 100 140 109 101 65 113 101 100 99 84 105 104

C:₩Users₩jimin.DESKTOP-8V2QQSQ₩source₩repos₩Project1₩Debug₩Project1.exe(프로세스 1556개)이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요.
```