Computer Programming - Final

考試時間 Test time 15:50 pm to 18:20 pm; 2.5 hours. 以投影時鐘為準 Using the projected clock as official time. 在答案卷上作答 Answer on the answer sheet. 每題一分 One point per problem. 全對才給分 No partial credit. 不倒扣 No penalty for wrong answers. 禁止外部資源（比照學測、分科） No external resources, like TOEFL.

[1] In C++, `int`, `float` are data types; they are like nouns. And `const` is like an adjective which describes a data type. In C++, what does `const int x = 10;` mean? (A) `x` is a constructor of an integer. (B) `x` is an integer that is constantly changing. (C) `x`'s value cannot be changed after initialization. (D) `x` is can only be used in functions declared with `const`.

[2] In C++, there are left values and right values. Left values are things that appears on the left side of =: they have memory addresses and can be read and written. Right values are things that can only be on the right side of =: they are mathematical values without a home in memory. Which of the following is a left value? (E) the result of `3 + 4` (F) the literal number `42` (G) an entry of array `A[3]` (H) a member variable of a class `obj.var` (I) the return value of a function `int fibo(int n);`

[3] What does adding `const` after a member function declaration guarantee? (J) The function is kept in the memory until the program ends. (K) the argument of the function must themselves be `const`. (L) the function promise that it will not modify member variables. (M) The function acts directly on memory and returns a constant value.

[4] `std::vector<t>` is actually an array `T[???]` under the hood. The size of the array is 1 in the beginning, and every time the array is full, a new array with twice the size is created, and the old elements are copied to the new array. How much time does one `push_back` of a `vector<int>` need, where n is the current size? Select all that apply. (N) $\Theta(1)$ every time, guaranteed (O) Sometimes $\Theta(1)$, but the worst case is $\Theta(\log n)$ (P) Sometimes $\Theta(1)$, but the worst case is $\Theta(n)$ (Q) $\Theta(1)$ on average (R) $\Theta(\log n)$ on average (S) $\Theta(n)$ on average

[5] In C++, templates are tools that allow users to write, say, multiple structures and functions with only one copy of codes. For instance `pair<int, float>` and `pair<string, string>` are two different types, but we only define `pair` once. Which statement about templates is correct? (T) templates are instantiated at runtime (U) templates cannot be defined in header files (V) templates must be explicitly instantiated for built-in types (W) templates are instantiated only when used with specific types (X) templates must be explicitly instantiated for user-defined classes

[6] What is the purpose of operator overloading? (Y) To provide faster implementations of operators over existing implementations. (Z) To reduce the memory usage by reusing existing operators. (A) To allow operators to work with different data types

[7] A data type is said to be *fundamental* if it behaves nicely. Here, *behaves nicely* can mean different things, but the most significant property is that when we say `FT y = x;`, we can copy the content of `x` to `y` by something like `memcpy(&y, &x, sizeof(FT));`. This is really nice because (i) we do not need to jump between pointers to get the content of `x`, and (ii) CPU usually optimizes `memcpy` very well. Which of the following are fundamental data types? (B) char (C) int (D) float (E) string (F) `pair<char, int>` (G) `array<float, 300>` (H) `tuple<string, string, string>`

[8] To the right you can see that `p_square` returns a pointer to the memory address of `b`. But `b` will disappear after the function ends. So the returned pointer is pointing to something no longer existing. What is the name we give to this kind of problem? (I) memory leak (J) heap overflow (K) dangling pointer (L) segmentation fault

```
int* p_square(int a) {
    int b = a*a;
    return &b;
}
```

[9] We know from experience that `string` does not have a fixed length. So it probably uses `new` and `delete` to control memory dynamically. It also behaves like an array so `s[2]` is the third character of `s`. Usually when we have two arrays `int A[100], B[100];` and we want to copy `A` to `B`, we cannot say `B = A;` because arrays do not support assignment. And even if in languages that allow `B = A;`, it usually only copies the pointer, not the contents. But in C++, we can say `string B = A;` and it works as expected. This behavior is called (M) lazy copy (N) deep copy (O) shallow copy (P) pointer copy

[10] How does C++ implement that? (Q) through operator overloading (R) through function overloading (S) through multiple inheritance (T) through template specialization

```
struct Rat {
    int age;
    string name;
    ...
    void setAge(int a) { ... }
}
```

[11] The accessability of age is (U) public (V) private (W) projected (X) protected (Y) prohibited

[12] Regarding methods in a `struct`, what is correct? (Z) only struct can have methods; class cannot (A) only class can have methods; struct cannot (B) both struct and class can have methods (C) neither struct nor class can have methods without `public:`.

[13] In concurrent programming, what is a data race? (D) Multiple threads write to the same memory without coordination. (E) Multiple threads compute the best-fit linear regression on the same data. (F) Multiple threads compete for allocating memory for their own local variables. (G) One thread writes to the end of a file while another reads from the beginning of that file. (H) Threads try to finish their computation as fast as possible because the last one needs to summarize the results.

[14] Which of the following are common mistakes people make programming that produces wrong results? (I) Lock a variable but forget to unlock it (J) Forget to lock a shared variable before reading it (K) Use too many threads and cause excessive context switching (L) Allocate memory with `new` but forget to `delete` it later

[15] In the supplymentary material `https://en.algorithmica.org/hpc/algorithms/matmul/`, which of the following are good optimizations for matrix multiplication? Select all that apply. (M) Unrolling loops (N) Cache sub-matrices (O) Using multiple threads (P) Single-data multiple-operation (Q) Using `std::vector` instead of raw arrays

[16] What is the recommended way to check if `x` is `None` in Python? (R) `if x:` (S) `if x is None:` (T) `if x == None:` (U) `if None == x:`

```
def f(l = [1,2,3]):
    l.append(l[-1]+1)
    print(l)
f()
f()
```

[17] The code to the right outputs `[1, 2, 3, 4]` and then `[1, 2, 3, 4, 5]`. Why? Select everything that contributes to this result. (V) Because append modifies the list in place (W) Because the function secretly calls itself recursively. (X) Because the assignment `list1 = list2` is a shallow copy. (Y) Because the default argument is created once at function definition time and reused every call. (Z) Because Python evaluates default arguments every time the function is executed, adding one extra element on each call.

[18] According to PEP 8, what is the recommended indentation level in Python code? (A) one tab per indentation level (B) 2 spaces per indentation level (C) 4 spaces per indentation level (D) any number of spaces or tabs as long as it is consistent within a project

```
UnknownFunction :: String -> Int
UnknownFunction "" = 0
UnknownFunction (first:rest) =
let (_, _, best) = foldl' step (first, 1, 1) rest
in best
where
    step :: (Char, Int, Int) -> Char -> (Char, Int, Int)
    step (prev_ch, curr_len, best) ch
    | ch == prev_ch =
        let curr_len' = curr_len + 1
            best' = max best curr_len'
        in (prev_ch, curr_len', best')
    | otherwise = (ch, 1, best)
```

[19] What language is this? (E) Go (F) Rust (G) Java (H) Haskell

[20] What is the purpose of this program??? (I) Compute fibonacci numbers. (J) Check if a string is a palindrome (K) Compute the longest repeated substring (L) Find the greatest common divisor of two numbers (M) Count the frequency of each character in a string (N) Generate all permutations of the characters of a string

[21] PEP 8's comment on line-length is as follows *The default wrapping in most tools disrupts the visual structure of the code, making it more difficult to understand. The limits are chosen to avoid wrapping in editors with the window width set to 80, even if the tool places a marker glyph in the final column when wrapping lines. Some web based tools may not offer dynamic line wrapping at all.* Beyond PEP 8, what is the root cause of limiting line width across different programming languages and communities? (O) Old screens only

support 80 columns. (P) Human eyes have difficulty reading long lines. (Q) Long lines are harder to parse by interpreters/compilers. (R) Old internet protocol only support 88 characters per packet. (S) Some editors and tools do not support dynamic line wrapping.

[22] According to PEP 8, which naming conventions are recommended? (T) `UPPERCASE` for classes (U) `UPPERCASE` for constants (V) `UPPERCASE` for functions (W) `UPPERCASE` for variables (X) `lowercase` for classes (Y) `lowercase` for constants (Z) `lowercase` for functions (A) `lowercase` for variables (B) `PascalCase` for classes (C) `PascalCase` for constants (D) `PascalCase` for functions (E) `PascalCase` for variables

[23] Which line are compliant with PEP 8 whitespace rules? (F) `x=i+1;` (G) `x = i+1;` (H) `x = i + 1;` (I) `func(a,b)` (J) `func (a,b)` (K) `{dog:2}` (L) `{dog: 2}` (M) `{dog : 2}`

[24] `\X` is going to be a number. How do I define `\Remember` so that the value of `\X` is saved, and later restored? Please do not use 3.14159 directly; this is just an example. We want to macro to work for any value of `\X`.

```
\def\X{3.14159}
\Remember\X
% some calculation that might change \X
\Recall\X
% now \X is 3.14159 again
```

[25] Throughout the entire semester, I use the terminal command to compile lecture examples. `???-?? -!!!=!!!!! -@a@@ -@e@@@a -@@e@a@@i@ -@@@a@o@ -@@o@@e@@io@ hello.cpp -o h && ./h`. What are the question marks?

[26] What are the exclamation marks?

[27] What are the at signs? One point per two options.

[28] For the homework `honai`, the size of the tower is a `cin`, not a command line argument. So naturally, TA needs to launch the program and press 9 and then `enter`. Or does he? In fact, in the auto-grader, 9 is part of the command line. What is that line?

[29] How to show first 20 line of `animal.cpp` in command line.

[30] What does the code to the right produce?

```
a = [1, 2, 3, 4]
b = [10, 20, 30]
c = [100, 200, 300, 400, 500]
result = []
for x, y, z in zip(a, b, c):
    result.append(x + y + z)
print(result)
```

[31] Do you remember the baby step giant step algorithm from the homework? We recommend BSGS to solve the following equation: $29^n \bmod 103 = 36$

[32] Do you remember RSA from the homework? Given public key $(n, e) = (97097, 7)$, the plaintext is $m = 89$, what is the ciphertext c?

[33] What is the private key (p, q, d)?

[34] What is the counterpart of `std::map` in Python?

[35] With `numpy` and `A,B=np.meshgrid(np.asarray([1,2,3]),np.asarray([4,5]))`, what is `A` and `B`.

[36] What does the code to the right produce?

```
reduce(lambda x, y:
    min(
        n for n in range(1, x * y + 1)
        if n % x == 0 and n % y == 0
    ),
    range(1, 10)
)
```

[37] Sort these by their invention year: C, C++, Python, SageMath, TeX, LaTeX. You only need to write the initial letters, and I will assume the second C is C++.

Long answer questions

The following questions require long answers. If the space is not enough, do not Fermat. Write your answer somewhere else and tell me where it is, like the expiry date of the drinks sold in convenience stores.

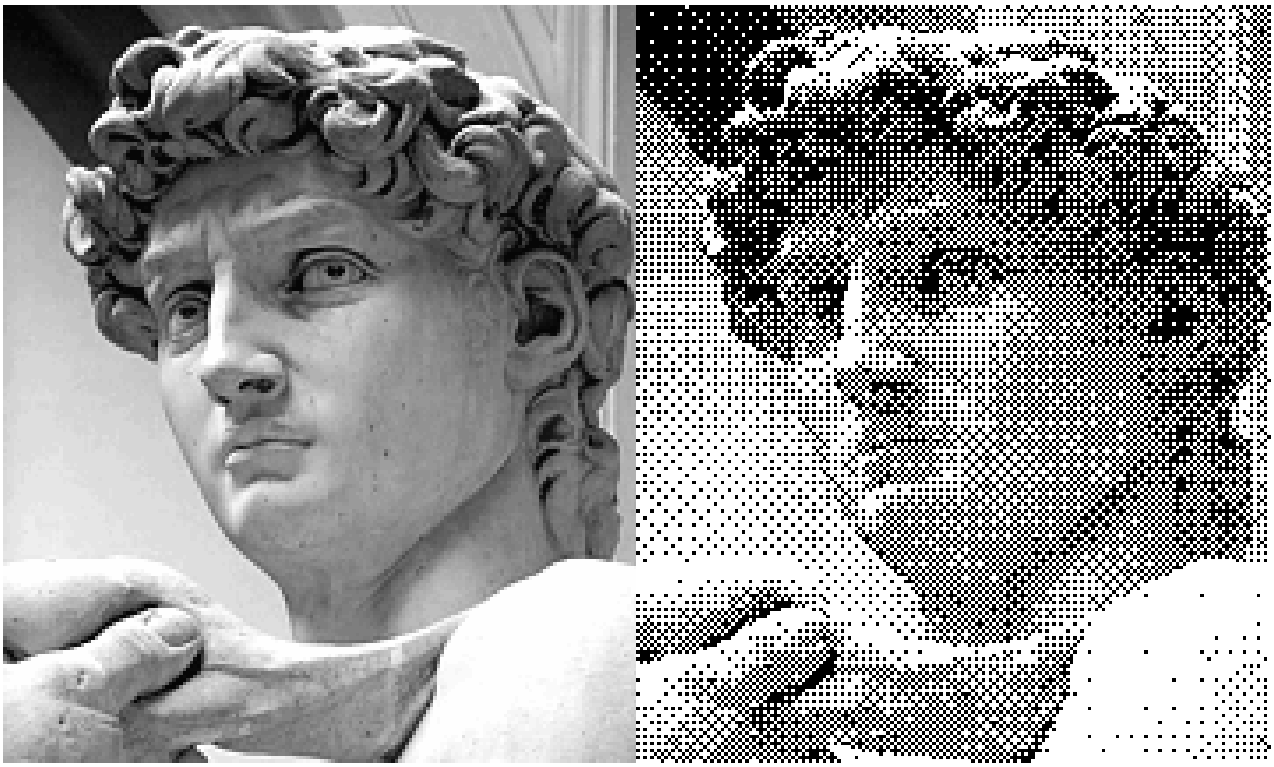
[38] Write a C++ code that prints the following sentence to the screen: *I do not cheat in this exam and, when the exam is over, I will take a photo of my answer sheet and upload it to ntu cool.*

[39] To the right is a usage of variadic template. Use the similar style to define `apply` so that `apply(x, f, g, h)` produces `h(g(f(x)))`.

```
template<typename T>
T sum(T one) { return one; }
template<typename T, typename ... TT>
T sum(T first, TT ... rest)
{ return first + sum(rest); }
```

[40] We all know that modern picture formats use RGB colors of 8 bits each. That is, the red component goes from 0 to 255, and so do the green and blue components. Therefore, each picture can use $256^3 = 16,777,216$ colors. However, some screens may support fewer colors. Does not mean that they cannot display pictures with more colors? Actually they can, by using a technique called ordered dithering.

```
int OD(uint8_t a, uint8_t b) {
    ...
    return o;
}
```



For simplicity, let's say that our gray-scale picture uses 5 colors: from 0 (pitch black) to 4 (blinding white). and our screen can only show two colors, 0 and 4. Consider this matrix

$$\begin{bmatrix} 0 & 2 \\ 3 & 1 \end{bmatrix}$$

and till it repeatedly to cover the entire picture.

$$\begin{bmatrix} 0 & 2 & 0 & 2 & 0 & 2 & \cdots \\ 3 & 1 & 3 & 1 & 3 & 1 & \cdots \\ 0 & 2 & 0 & 2 & 0 & 2 & \cdots \\ 3 & 1 & 3 & 1 & 3 & 1 & \cdots \\ 0 & 2 & 0 & 2 & 0 & 2 & \cdots \\ 3 & 1 & 3 & 1 & 3 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \end{bmatrix}$$

Now a gary-scale value $v \in \{0, 1, 2, 3, 4\}$ will be displayed as white if it is greater than the corresponding value in the above matrix, and black otherwise. For if we have 17 levels of gray-scale, we can use a 4×4 matrix like this:

$$\begin{bmatrix} 0 & 8 & 2 & 10 \\ 12 & 4 & 14 & 6 \\ 3 & 11 & 1 & 9 \\ 15 & 7 & 13 & 5 \end{bmatrix}$$

Finish the C++ code that helps generate the 16×16 matrix for 257 levels of gray-scale.

```
int TM(uint16_t a) {
    ...
    return b;
}
```

[41] The Thur–Morse sequence is defined as: $t_{2n} = t_n$, and $t_{2n+1} = 1 - t_n$ with initial term $t_0 = 0$. Using the Thue–Morse sequence for two-choice scheduling is fair because every prefix is nearly perfectly balanced: the counts of 0s and 1s differ by at most 1, and 0s take a lead about 50% of the time. It also avoids long streaks (no 000 or 111), preventing one side from getting several consecutive turns, which reduces perceived bias while remaining deterministic and simple to generate. Finish the function `TM`.

```
template<class Iter, class Compare>
void sort(Iter first, Iter last, Compare comp) {...}
```

[42] To the right is a generic sort function. To use it, I need to write `sort(Container.begin(), Container.end(), compare)`. But I don't want to repeat `Container` two times. Write a `sort` that takes only `Container` and `compare`

```
sort(VI.begin(), VI.end(),
[](auto a, auto b) {
    ???
});
```

[43] in `sort`, the function `compare(a, b)` should return `true` if the first argument should be before the second argument. Implement a compare function that sorts `vector<int16_t> VI` using the following order: take the binary representation of an `int16_t`, reverse it, and compare the reversed representation. In other words, 11 will be treated as if it is `1101 0000 0000 0000`.

```
class SWOT {
    public:
    string S, W, O, T;
}
ostream &operator<<(ostream &dcard, SWOT ntuee) {
    ...
    return dcard;
}
```

[44] In a business class we learn about the SWOT analysis. SWOT stands for strengths, weaknesses, opportunities, and threats. It is usually represented in a 2×2 matrix in this order $\left[\begin{smallmatrix} S & W \\ O & T \end{smallmatrix}\right]$. Implementing the printing operator for `SWOT` so that each column's width is the maximum width of the two entries plus two padding spaces, and columns are separated by vertical bars. The following is an example output:

```
| Many buildings and labs | A lot of homeworks               |
| AI explosion            | Nanyang Technological University |
```

Hint: the length of a `string` is `.size()`.

```
while True:
    plant(???)
    use_item(???, 1024)
    while get_entity_type() != ???:
        ### move along left wall ###
    harvest()
```

[45] In the video game Farmer Was Replaced, I use this code to find treasures in a maze. What are the question marks?

[46] Finish the code that moves along the left wall.

[47] `change_hat(Hats.Dinosaur_Hat)` enters the dinosaur mode, and `change_hat(Hats.Straw_Hat)` exits it. Write a code that can fill the entire 32-by-32 farm with dinosaur body.

[48] To get one more point, improve your dinosaur code so that it restarts automatically when the farm is full.

[49] This exam paper uses `\usepackage[???]{geometry}`. All numbers are from the E6 preferred number series.

[50] What does `plot(x^2 * sin(1/x), -1, 1)` produce in SageMath? This function is one of the classic example of functions that is differentiable everywhere but not continuously differentiable.

Computer Programming - Final

Name (zh or en) Student ID

[1] [2] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12]

[13] [14] [15] [16] [17] [18] [19] [20] [21] [22] [23] [24]

[25] [26] [27] [28] [29] [30]

[31] [32] [33] [34] [35] [36] [37]

[38]

[39]

[40]

[41]

[42]

[43]

[44]

[45]

[46]

[47]

[48]

[49]

[50]