Computer Programming - Final Mock

考試時間 Test time 15:50 pm to 18:10 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.

```
class Rabbit {
    int age;
    string name;
public:
    // constructors here
    // get_age() here
    // rename() here
}
```

[1] By default, class members are (A) protected (B) private (C) public (D) friend

[2] By default, struct members are (E) protected (F) private (G) public (H) friend

[3] Define the default constructor that sets the age to `0` and the name to `unnamed rabbit`.

[4] Define a constructor that sets the age to `0` and the name to the argument.

[5] Define the getter function of `age`.

[6] Define the setter function `rename` that takes one string as argument; if the string contains only `a`–`z` (all lowercase), then rename. Otherwise return false.

```
class Dog { public:
    int age;
    ...
}
class Chiwawa : public Dog { public:
    int dB;
    ...
}
```

[7] If `p1` is a pointer to a dog, how does `p1->age` work? (I) The compiler replaces it with `p1.(*age)`. (J) The runtime searches for `age` in the dictionary of of the dog. (K) The compiler replaces it with `*(p1 + shift)` for some number `shift`.

[8] If the type of `p2` is pointer-to-dog, but it points to a chiwawa, how does `p2->age` work? (L) The runtime cast `*p2` to a dog first and the problem reduces to the previous question. (M) The runtime searches for `age` in the dictionary of of the chiwawa. (N) The compiler replaces it with `*(p2 + shift)` for the same `shift` as the last question. (O) The compiler replaces it with `*(p2 + shift)` for some different `shift`. (P) This does not work.

[9] If the type of `p3` is a pointer-to-chiwawa, but it points to a dog, how does `p3->age` work? (Q) This is not possible. (R) The runtime cast `*p3` to chiwawa using the default constructor. (S) The compiler replaces it with `*(p3 + shift)` for the same number as the last question. (T) The compiler replaces it with `*(p3 + shift)` for some different number `shift`.

[10] If the type of `p4` is a pointer-to-dog, but it points to a chiwawa, how does `p4->dB` work? (U) The runtime cast `p4` to pointer-to-chiwawa and perform `->dB` (V) The runtime searches for `dB` in the dictionary of of the chiwawa. (W) The compiler replaces it with `*(p4 + shift)` for some different number `shift`. (X) This does not work.

[11] If you answer to the previous question implies that the compiler knows that p4 points to a chiwawa, why does it know? If your answer implies that the compiler does not know, how to fix the code so that it knows?

[12] Which of the following are the benefit of using array of structures: `Dog HuangBaBa[100]`? (Y) Better cache locality when increasing all dogs ages. (Z) Encapsulation and readable code: easy to pass `Dog*` to functions. (A) Easier to sort dogs by age or name as we only modify one array. (B) Uses less memory because this avoids padding between two arrays.

[13] Which of the following is the benefit of using structures of arrays: `int ages[100]; string names[100];`? (C) Enables SIMD/vectorization for single-field operations. (D) Uses less memory because this avoids padding between members. (E) Easier to sort dogs by age or name as we only modify one array. (F) Encapsulation and readable code: it suffices to pass index `i` to functions.

```
int y = 2;
@@@ f = [=y](auto x) { return x * y; }
### g = [&y](auto x) { return x + y; }
y = 1;
```

[14] What should @@@ be?

[15] What should ### be?

[16] What is `f(g(2))`?

[17] Using composition of the form `f(g(...(g(f(0)))...))` to produce `101`.

```
auto pi = make_pair(3.14, "pi");
auto e = make_pair(2.71, "e");
auto [a, b] = pi;
auto [c, d] = e;
```

[18] Another way to obtain `a` is `auto a = pi.???;`

[19] Another way to obtain `b` is `auto b = pi.???;`

[20] The function template `make_pair` is defined in the standard library. But how is it defined? You may assume that the structure template `pair` is already defined. You also dont need to worry about `const` or `&` (pass by reference thing).

[21] How to use `decltype` to make `DS` the type of `pi`?

[22] Overload the `+` operator for `DS` so that `pi + e` produces `make_pair(5.85, "pi + e")`.

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

[23] 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)))`.

[24] Let $t_0, t_1, t_2, \ldots$ be the Thur–Morse sequence defined as:

$$t_0 = 0,\ t_{2n} = t_n,\ t_{2n+1} = 1 - t_n.$$

Let `Car` be a class. Overload the `+` operator for `queue<Car>` so that the jth element of the `q1 + q2` is from `q1` if $t_j = 0$ and from `q2` if $t_j = 1$. Note that cars from the same queue should remain in the same order.

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

[25] For something like `vector<vector<int>> VVI = { {1,2,3}, {4,5}, {6} };` how to sort it so that shorter vectors come first, and for vectors of the same length, lower sum comes first, and for vectors of the same length and sum, using dictionary order?

[26] Overload the `()` operator for `vector<T>` so that `v(3)` returns `v[3]` and `v(-5)` returns `v[v.size() - 5]`.

[27] Overload the `[]` operator for `vector<T>` so that `v[vector<int>2, 4, 8]` returns `vector<T>v[2], v[4], v[8]`.

[28] An RSA public key is `p * q = 91; e = 5`, what is the private key `d`?

```
array<bool, 127> decode(array<bool, 127> y) {
    ???
}
```

[29] A Hamming codeword is an `array<bool, 127> x;` such that $xH = 0$, where H is the matrix where the jth row is the binary representation of `j`; so the first row is `0000001` and the last row is `1111111`. Given another `array<bool, 127> y`, find the codeword `x` that differs from `y` by at most one bit.

```
A = [1, 2, 3]
B = "w x y z"
for a, b in zip(A, B):
    print(a, b)
```

[30] What is the first line of the output?

[31] What is the second line of the output?

[32] What is the fourth line of the output? (If there is no fourth line, write NONE).

[33] What is `list(map(lambda x: B[x], A))`?

[34] What is `reduce(lambda x, y: max(x, y), B, B[0])`?

[35] What is `filter(lambda x: x & 1, B)`?

[36] In the video game *The Farmer Was Replaced*, how to fertilize the entire field exactly once?

```
\def\rota#1#2#3{#2#3#1}
\def\rotb#1#2#3{#3#1#2}
\def\refl#1#2#3{#3#2#1}
```

[37] What is `\rota 123456789`?

[38] What is `\rotb\rota 123456789`?

[39] What is `\expandafter\refl\refl 123456789`?

[40] What is `\expandafter\expandafter\expandafter\refl\refl 123456789`?

```
\def\two#1{tour}
\def\to{\two\two}
\edef\too{\two\two}
\edef\tool{\noexpand\two\noexpand\two}
\edef\toll{\noexpand\two\to}
\edef\toll{\toll}
\def\two#1{tall}
\to \\ \too \\ \tool \\ \toll
```

[41] What is the first line of the output?

[42] What is the second line of the output?

[43] What is the third line of the output?

[44] What is the fourth line of the output?

[45] Which Ubuntu version is used in autograder?

[46] Which g++ version is used in autograder?

```
let add = |a: i32, b: i32| -> i32 { a * b };
let x = 5;
let y = 7;
let result = add(x, y);
println!("The sum is: {}", result);
```

[47] What is the output?

[48] What is `i32`?

[49] What is this language? (G) C (H) C++ (I) Rust (J) python (K) Haskell

[50] When did this language reach 1.0? (L) 1975 (M) 1995 (N) 2015 (O) 2025

Computer Programming - Final Mock

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]