



Σημείωμα

Για όλα τα προβλήματα:

- Τα όρια είναι διαθέσιμα στη σελίδα "Overview" στο σύστημα του διαγωνισμού
- Υπάρχει ένα πακέτο συνημμένου που μπορείτε να κατεβάσετε από το σύστημα του διαγωνισμού, το οποίο περιέχει δείγματα βαθμολογητών, δείγματα υλοποιήσεων, παραδείγματα αρχείων ελέγχου και scripts μεταγλώττισης και εκτέλεσης.
- Μπορείτε να κάνετε έως και 50 υποβολές για κάθε πρόβλημα, και πρέπει να υποβάλετε ακριβώς ένα αρχείο σε κάθε υποβολή.
- Όταν ελέγχετε τα προγράμματά σας με το υπόδειγμα βαθμολογητή, η είσοδό σας θα πρέπει να συμφωνεί με τη μορφή και τους περιορισμούς από την εκφώνηση του προβλήματος, διαφορετικά ενδέχεται να προκύψουν απροσδιόριστες συμπεριφορές
- Στις εισόδους του βαθμολογητή, κάθε δύο διαδοχικά σύμβολα σε μια γραμμή χωρίζονται από ένα μόνο κενό διάστημα, εκτός εάν ορίζεται ρητά άλλη μορφή.
- Όταν ελέγχετε τον κώδικά σας τοπικά στον υπολογιστή σας, συνιστούμε να χρησιμοποιείτε scripts στα πακέτα συνημμένων. Λάβετε υπόψη ότι χρησιμοποιούμε τον `-std=gnu++17` μεταγλωττιστή.
- Εάν δεν μπορείτε να υποβάλετε στο CMS, μπορείτε να χρησιμοποιήσετε το εργαλείο «ioisubmit» για να αποθηκεύσετε τον κώδικά σας για αξιολόγηση μετά το τέλος του διαγωνισμού.
 - Εκτελέστε το `ioisubmit <task_shortname> <source_file>` στον φάκελο με το `<source_file>`.
 - Ζητήστε από ένα μέλος της επιτροπής να φωτογραφίσει την έξοδο του «ioisubmit». Η υποβολή σας δεν θα ληφθεί υπόψη εάν δεν έχει γίνει αυτό το βήμα.
 - Εάν διαγωνίζεστε διαδικτυακά, ζητήστε από τον επιβλέποντά σας να φωτογραφίσει την έξοδο του «ioisubmit» και να στείλει τη φωτογραφία στους διοργανωτές.

Σύμβαση

Οι εκφωνήσεις των προβλημάτων προσδιορίζουν τις υπογραφές χρησιμοποιώντας τα γενικευμένα ονόματα τύπων `void`, `bool`, `int`, `int[]` (πίνακας), και `union(bool, int[])`.

Στη C++, οι βαθμολογητές χρησιμοποιούν κατάλληλους τύπους δεδομένων ή υλοποιήσεις, όπως αναφέρονται παρακάτω

void	bool	int	int[]
void	bool	int	std::vector<int>

union(bool, int[])	length of array a
std::variant<bool, std::vector<int>>	a.size()

Στη C++, το `std::variant` ορίζεται στο αρχείο κεφαλίδας `<variant>`

Μια μέθοδος με τύπο επιστροφής `std::variant<bool, std::vector<int>>` μπορεί να επιστρέψει είτε `bool` είτε `std::vector<int>`. Το παρακάτω δείγμα κώδικα δείχνει τρία παραδείγματα χρήσης συναρτήσεων που επιστρέφουν `std::variant`.

```
std::variant<bool, std::vector<int>> foo(int N) {
    return N % 2 == 0;
}
std::variant<bool, std::vector<int>> goo(int N) {
    return std::vector<int>(N, 0);
}
std::variant<bool, std::vector<int>> hoo(int N) {
    if (N % 2 == 0) {
        return false;
    }
    return std::vector<int>(N, 0);
}
```