МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

**Лабораторная работа №5**

по дисциплине: Основы программирования

тема: «Циклы»

Выполнил: ст. группы ПВ-223

Пахомов Владислав Андреевич

Проверили:

Притчин Иван Сергеевич

Черников Сергей Викторович

Код-ревьер: ст. группы ПВ-223

Голуцкий Георгий Юрьевич

Белгород 2022 г.

**Лабораторная работа № 5**

**Содержание отчёта:**

* Тема лабораторной работы
* Цель лабораторной работы
* Решения задач
  + Название задачи
  + Блок-схема (для задач с звездочкой)
  + Исходный код
  + Скриншот с codeforces с указанием вердикта тестирующей системы
* Код-ревью
* Работа над ошибками (код-ревью)
* Вывод по работе.

**Тема лабораторной работы:** Циклы

**Цель лабораторной работы:** закрепление навыков написания циклических алгоритмов

**Решения задач:**

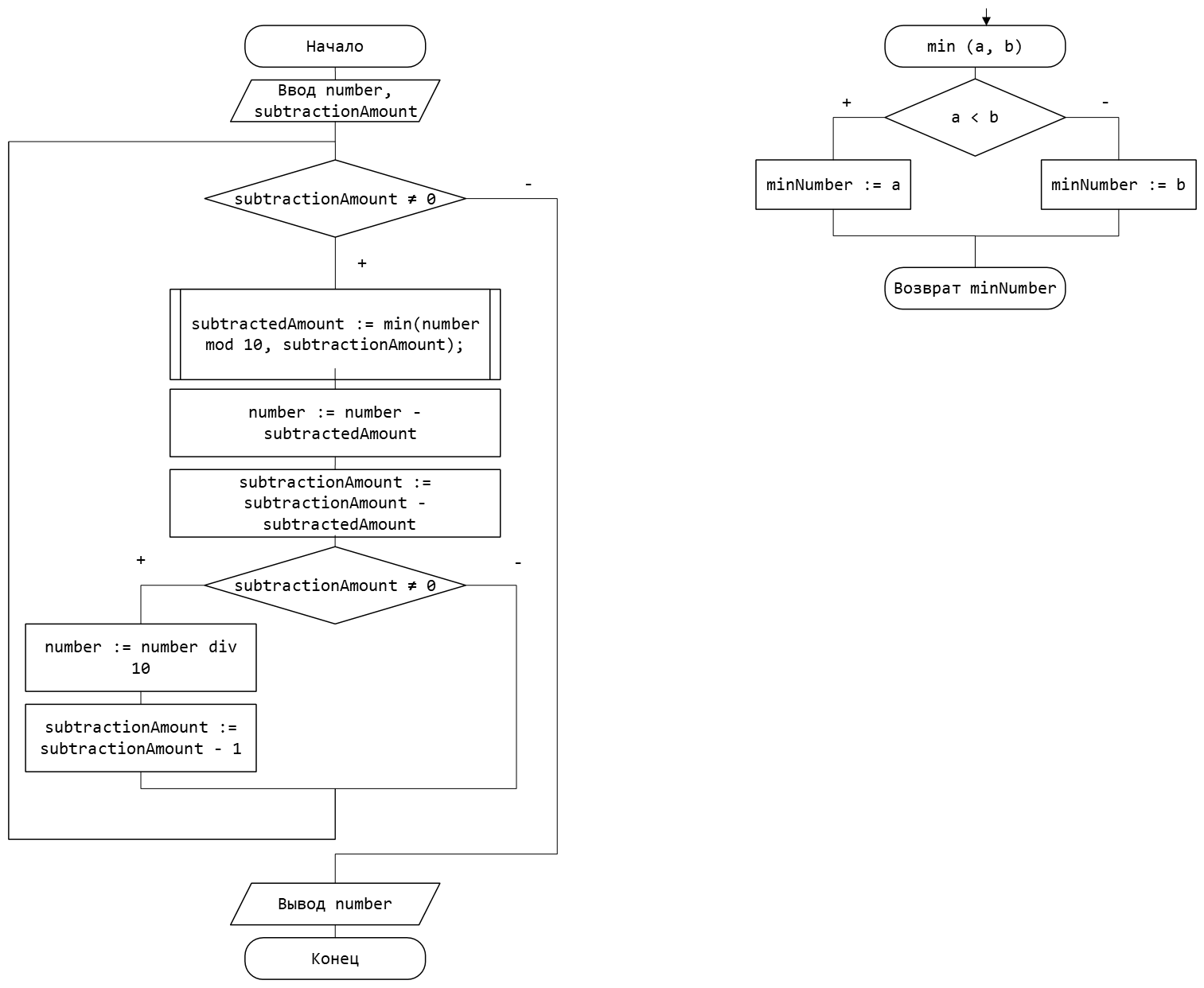
1. Команда (231A)

#include <stdio.h>  
  
int main() {  
 int taskAmount;  
 scanf("%d", &taskAmount);  
  
 int implementedProblems = 0;  
  
 for (int currentTask = 0; currentTask < taskAmount; currentTask++) {  
 int viewPetya, viewVasya, viewTonya;  
 scanf("%d %d %d", &viewPetya, &viewVasya, &viewTonya);  
  
 int viewTotal = viewPetya + viewVasya + viewTonya;  
 implementedProblems += viewTotal / 2;  
 }  
  
 printf("%d", implementedProblems);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/231A.c>



1. \* Неправильное вычитание (977A)

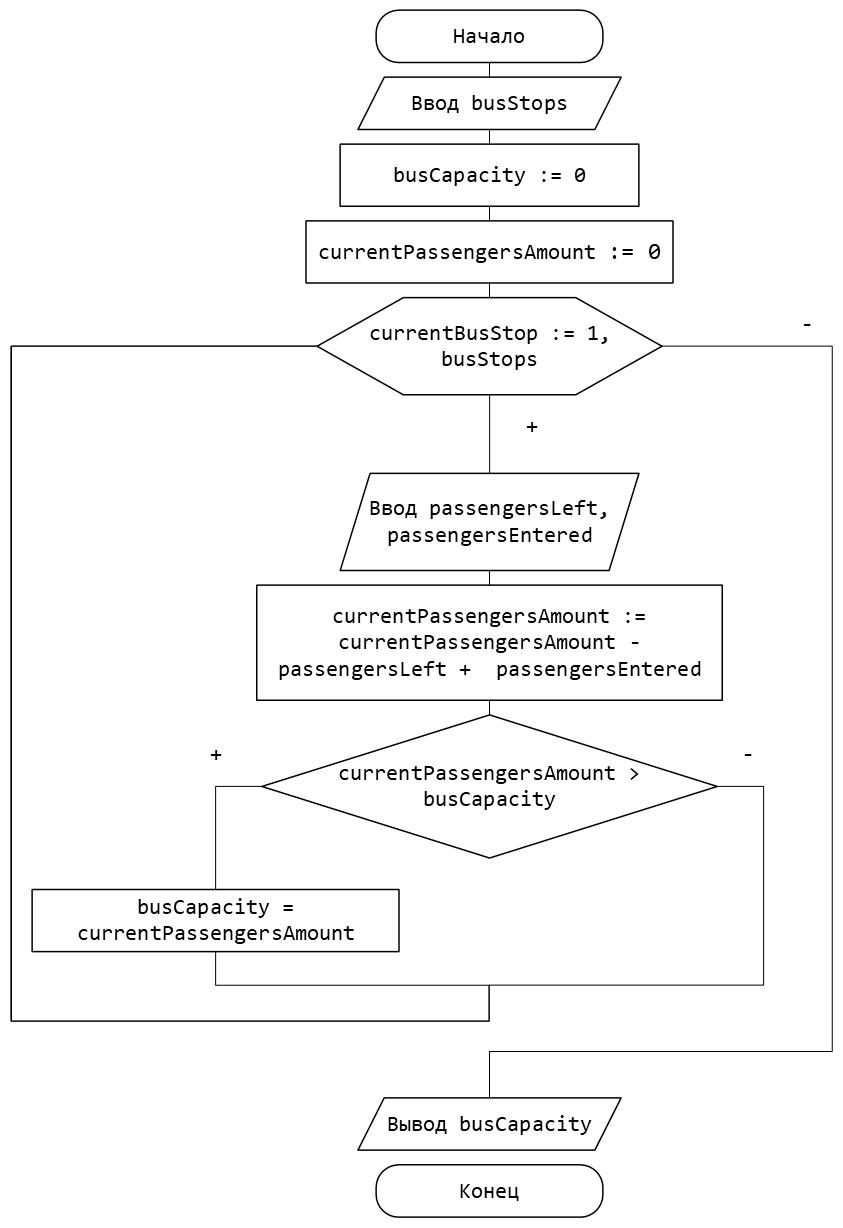


#include <stdio.h>  
  
long long min(long long a, long long b) {  
 return a < b ? a : b;  
}  
  
int main() {  
 int number, subtractionAmount;  
 scanf("%d %d", &number, &subtractionAmount);  
  
 while (subtractionAmount) {  
 int subtractedAmount = min(number % 10, subtractionAmount);  
 number -= subtractedAmount;  
 subtractionAmount -= subtractedAmount;  
  
 if (subtractionAmount) {  
 number /= 10;  
 subtractionAmount--;  
 }  
 }  
  
 printf("%d", number);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/977A.c>



1. \* Трамвай (116A)



#include <stdio.h>  
  
int main() {  
 int busStops;  
 scanf("%d", &busStops);  
  
 int busCapacity = 0;  
 int currentPassengersAmount = 0;  
  
 for (int currentBusStop = 0; currentBusStop < busStops; currentBusStop++) {  
 int passengersLeft, passengersEntered;  
 scanf("%d %d", &passengersLeft, &passengersEntered);  
  
 currentPassengersAmount += passengersEntered - passengersLeft;  
  
 if (currentPassengersAmount > busCapacity)  
 busCapacity = currentPassengersAmount;  
 }  
  
 printf("%d", busCapacity);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/116A.c>



1. Ваня и забор (677A)

#include <stdio.h>  
  
int main() {  
 int friendsAmount, fenceHeight;  
 scanf("%d %d", &friendsAmount, &fenceHeight);  
  
 int roadWidth = 0;  
  
 for (int currentFriend = 0; currentFriend < friendsAmount; currentFriend++) {  
 int friendHeight;  
 scanf("%d", &friendHeight);  
  
 roadWidth += 1 + (friendHeight > fenceHeight);  
 }  
  
 printf("%d", roadWidth);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/677A.c>

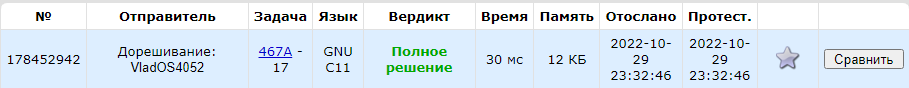


1. Юра и заселение (467A)

#include <stdio.h>  
  
int main() {  
 int roomsAmount;  
 scanf("%d", &roomsAmount);  
  
 int roomsAvailableForMovingIn = 0;  
  
 for (int currentRoom = 0; currentRoom < roomsAmount; currentRoom++) {  
 int roomLivingPeople, roomAccommodatePeople;  
 scanf("%d %d", &roomLivingPeople, &roomAccommodatePeople);  
  
 roomsAvailableForMovingIn += (roomAccommodatePeople –

roomLivingPeople) >= 2;  
 }  
  
 printf("%d", roomsAvailableForMovingIn);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/467A.c>



1. Выбор команд (432A)

#include <stdio.h>  
  
#define **PARTICIPATION\_MAX\_AMOUNT** 5  
#define **TEAM\_PARTICIPANTS** 3  
  
int main() {  
 int pupilAmount, requiredParticipationAmount;  
 scanf("%d %d", &pupilAmount, &requiredParticipationAmount);  
  
 int pupilSuitableParticipationAmount = 0;  
  
 for (int currentPupil = 0; currentPupil < pupilAmount; currentPupil++) {  
 int pupilParticipationAmount;  
 scanf("%d", &pupilParticipationAmount);  
  
 pupilSuitableParticipationAmount += (**PARTICIPATION\_MAX\_AMOUNT** –

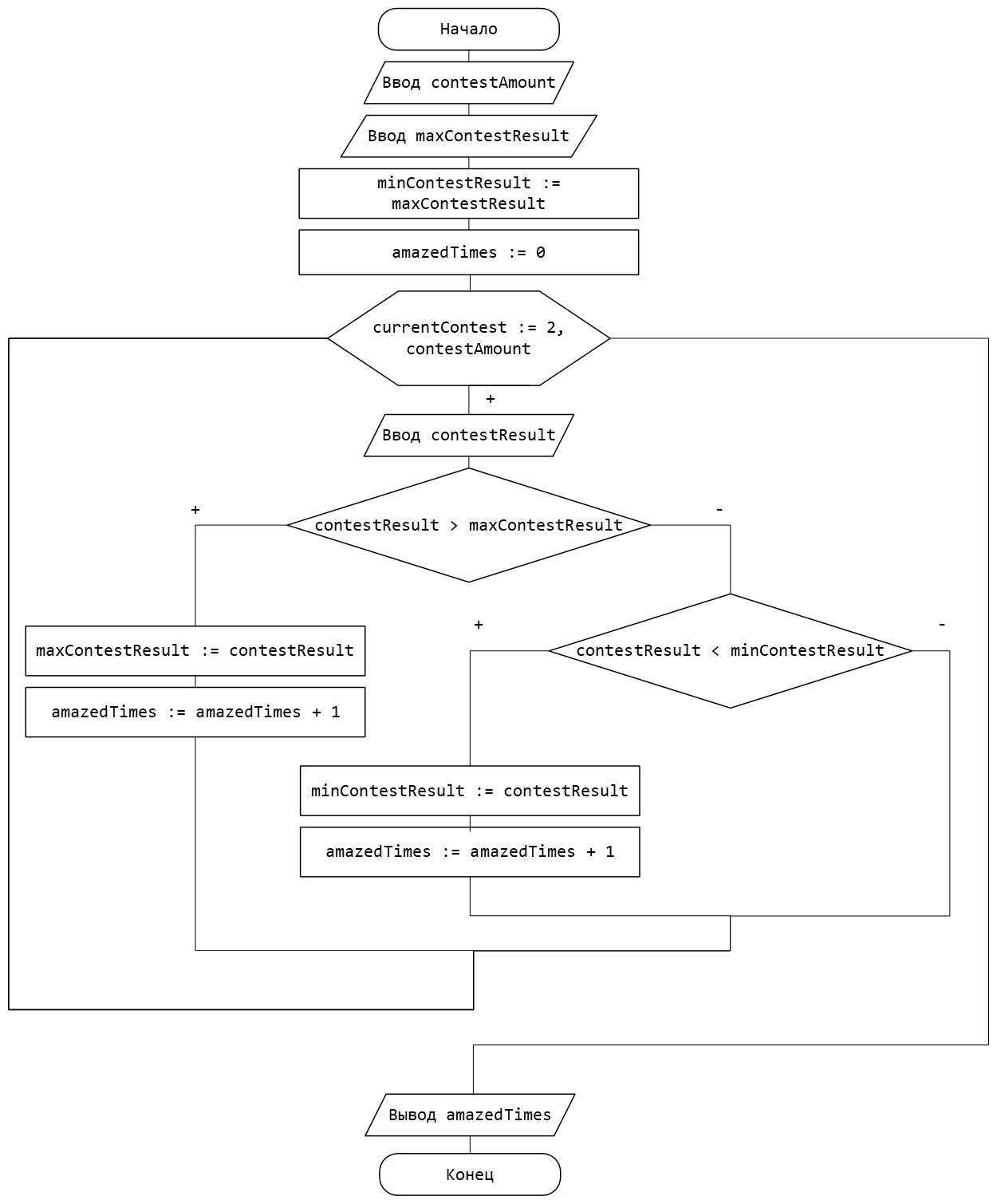
pupilParticipationAmount) >=

requiredParticipationAmount;  
 }  
  
 int teamsAmount = pupilSuitableParticipationAmount / **TEAM\_PARTICIPANTS**;  
  
 printf("%d", teamsAmount);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/432A.c>



1. \* I\_love\_%username% (155A)



#include <stdio.h>  
  
int main() {  
 int contestAmount;  
 scanf("%d", &contestAmount);  
  
 int maxContestResult;  
 scanf("%d", &maxContestResult);  
  
 int minContestResult = maxContestResult;  
 int amazedTimes = 0;  
  
 for (int currentContest = 1; currentContest < contestAmount; currentContest++) {  
 int contestResult;  
 scanf("%d", &contestResult);  
  
 if (contestResult > maxContestResult) {  
 maxContestResult = contestResult; *// :D* amazedTimes++;  
 } else if (contestResult < minContestResult) {  
 minContestResult = contestResult; *// D:* amazedTimes++;  
 }  
 }  
  
 printf("%d", amazedTimes);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/155A.c>



1. Нечётное множество (1542A)

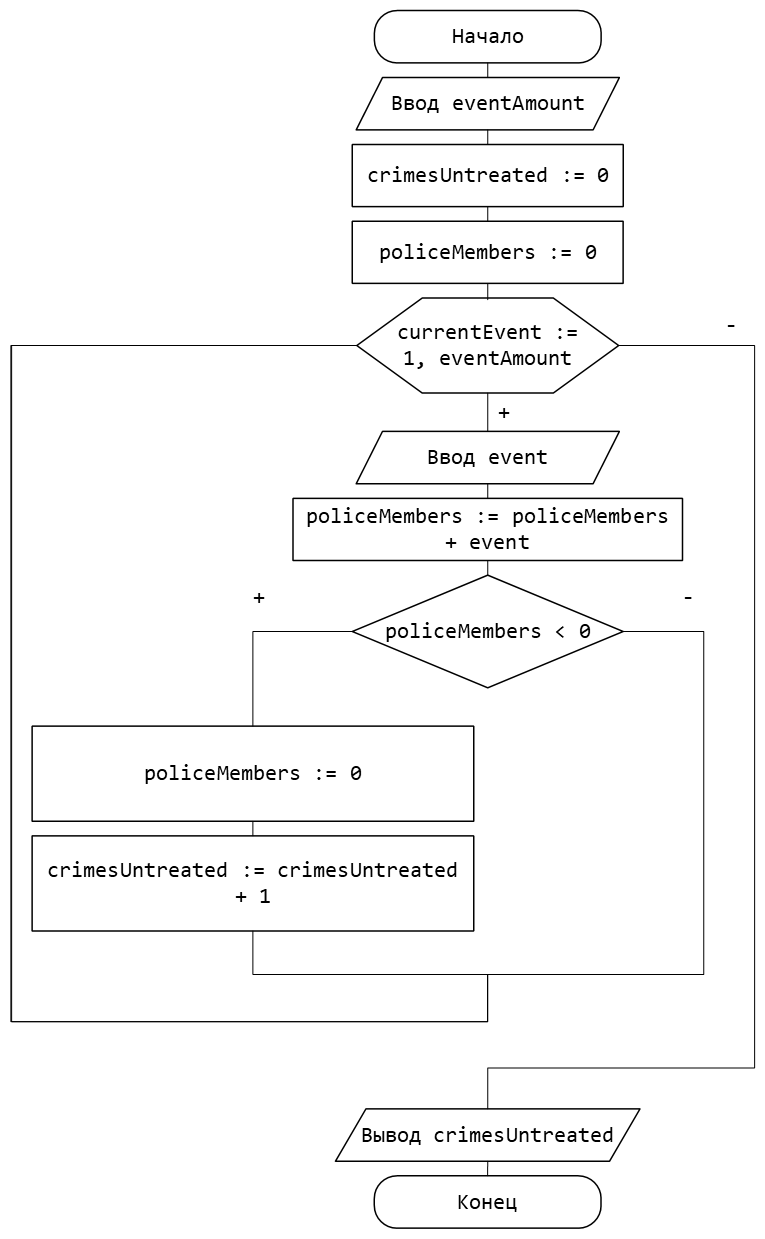
#include <stdio.h>  
  
int main() {  
 int taskAmount;  
 scanf("%d", &taskAmount);  
  
 for (int currentTask = 0; currentTask < taskAmount; currentTask++) {  
 int numbersAmount;  
 scanf("%d", &numbersAmount);  
  
 int oddCount = 0;  
  
 for (int currentNumber = 0; currentNumber < 2 \* numbersAmount;

currentNumber++) {  
 int number;  
 scanf("%d", &number);  
  
 oddCount += number % 2;  
 }  
  
 if (oddCount == numbersAmount)  
 printf("Yes\n");  
 else  
 printf("No\n");  
 }  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/1542A.c>



1. \* Полицейские-рекруты (427A)



#include <stdio.h>  
  
int main() {  
 int eventAmount;  
 scanf("%d", &eventAmount);  
  
 int crimesUntreated = 0;  
 int policeMembers = 0;  
  
 for (int currentEvent = 0; currentEvent < eventAmount; currentEvent++) {  
 int event;  
 scanf("%d", &event);  
  
 policeMembers += event;  
  
 if (policeMembers < 0) {  
 policeMembers = 0;  
 crimesUntreated++;  
 }  
 }  
  
 printf("%d", crimesUntreated);  
  
 return 0;  
}

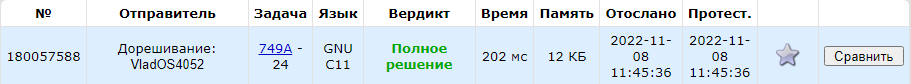
Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/427A.c>



1. Задача Бахгольда (749A)

#include <stdio.h>  
  
int main() {  
 int number;  
 scanf("%d", &number);  
  
 int primeNumbersAmount = number / 2;  
 int isOdd = number % 2;  
  
 printf("%d\n", primeNumbersAmount);  
  
 for (int i = isOdd; i < primeNumbersAmount; i++)  
 printf("2 ");  
  
 if (isOdd)  
 printf("3");  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/749A.c>



1. Мишка и старший брат (791A)

#include <stdio.h>  
  
int main() {  
 int weightLimak, weightBob;  
 scanf("%d %d", &weightLimak, &weightBob);  
  
 int fullYearsToBecomeLarger = 0;  
  
 while (weightLimak <= weightBob) {  
 weightLimak \*= 3;  
 weightBob \*= 2;  
 fullYearsToBecomeLarger++;  
 }  
  
 printf("%d", fullYearsToBecomeLarger);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/791A.c>



12. Открытки для друзей (1472A)

#include <stdio.h>  
  
int main() {  
 int taskAmount;  
 scanf("%d", &taskAmount);  
  
 for (int currentTask = 0; currentTask < taskAmount; currentTask++) {  
 int paperWidth, paperHeight, requiredPostcardsAmount;  
 scanf("%d %d %d", &paperWidth, &paperHeight, &requiredPostcardsAmount);  
  
 int maxCardsAmount = 1;  
  
 while (paperWidth % 2 == 0) {  
 paperWidth /= 2;  
 maxCardsAmount \*= 2;  
 }  
  
 while (paperHeight % 2 == 0) {  
 paperHeight /= 2;  
 maxCardsAmount \*= 2;  
 }  
  
 if (maxCardsAmount >= requiredPostcardsAmount)  
 printf("YES\n");  
 else  
 printf("NO\n");  
 }  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/1472A.c>



13. Ваня и кубики (492A)

#include <stdio.h>  
  
int main() {  
 int cubesTotal;  
 scanf("%d", &cubesTotal);  
  
 int maxHeight = 0;  
 int cubesCurrentTotal = 0;  
  
 while (cubesCurrentTotal <= cubesTotal) {  
 maxHeight++;  
 cubesCurrentTotal += (maxHeight \* (maxHeight + 1)) / 2;  
 }  
  
 maxHeight--;  
  
 printf("%d", maxHeight);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/492A.c>



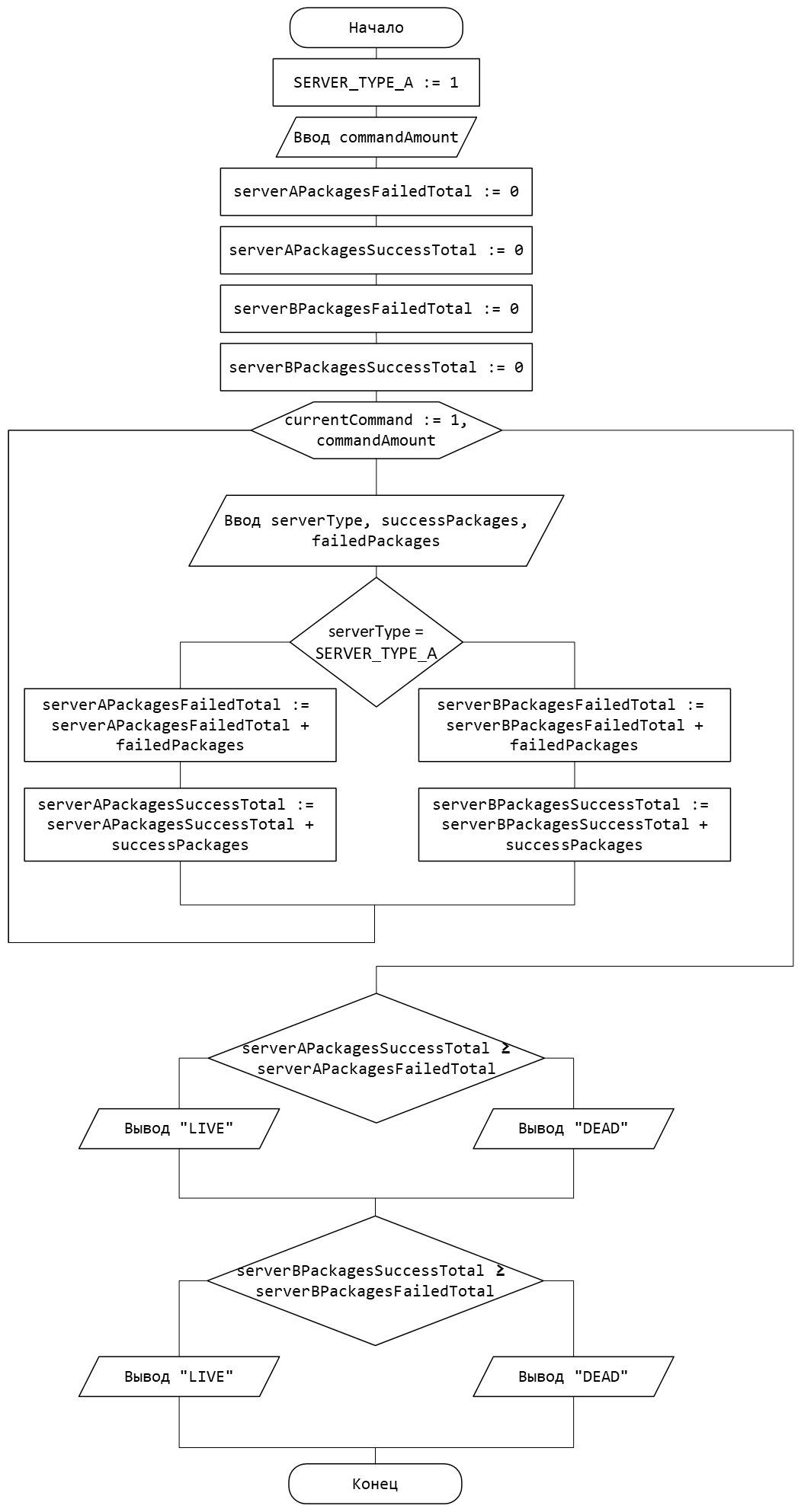
14. Сайт отзывов (1511A)

#include <stdio.h>  
  
#define **USER\_TYPE\_DISLIKE** 2  
  
int main() {  
 int taskAmount;  
 scanf("%d", &taskAmount);  
  
 for (int currentTask = 0; currentTask < taskAmount; currentTask++) {  
 int usersAmount;  
 scanf("%d", &usersAmount);  
  
 int maxLikes = 0;  
  
 for (int currentUser = 0; currentUser < usersAmount; currentUser++) {  
 int userType;  
 scanf("%d", &userType);  
  
 maxLikes += userType != **USER\_TYPE\_DISLIKE**;  
 }  
  
 printf("%d\n", maxLikes);  
 }  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/1511A.c>



15. \* Системный администратор (245A)



#include <stdio.h>  
  
#define **SERVER\_TYPE\_A** 1  
  
int main() {  
 int commandAmount;  
 scanf("%d", &commandAmount);  
  
 int serverAPackagesFailedTotal = 0;  
 int serverAPackagesSuccessTotal = 0;  
 int serverBPackagesFailedTotal = 0;  
 int serverBPackagesSuccessTotal = 0;  
  
 for (int currentCommand = 0; currentCommand < commandAmount; currentCommand++) {  
 int serverType, successPackages, failedPackages;  
 scanf("%d %d %d", &serverType, &successPackages, &failedPackages);  
  
 if (serverType == **SERVER\_TYPE\_A**) {  
 serverAPackagesFailedTotal += failedPackages;  
 serverAPackagesSuccessTotal += successPackages;  
 } else {  
 serverBPackagesFailedTotal += failedPackages;  
 serverBPackagesSuccessTotal += successPackages;  
 }  
 }  
  
 if (serverAPackagesSuccessTotal >= serverAPackagesFailedTotal)  
 printf("LIVE\n");  
 else  
 printf("DEAD\n");  
  
 if (serverBPackagesSuccessTotal >= serverBPackagesFailedTotal)  
 printf("LIVE\n");  
 else  
 printf("DEAD\n");  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/245A.c>

16. Покупка еды (1296B)

#include <stdio.h>  
  
#define **CASHBACK\_TOTAL\_PARTS** 10  
#define **CASHBACK\_PART** 1  
  
int main() {  
 int purchaseAmount;  
 scanf("%d", &purchaseAmount);  
  
 for (int currentPurchase = 0; currentPurchase < purchaseAmount;

currentPurchase++) {  
 long long moneyTotal;  
 scanf("%lld", &moneyTotal);  
  
 long long moneySpentTotal = 0;  
  
 while (moneyTotal > **CASHBACK\_TOTAL\_PARTS** - **CASHBACK\_PART**) {  
 long long moneyCashback = moneyTotal / **CASHBACK\_TOTAL\_PARTS**;  
 moneyTotal = (moneyTotal % **CASHBACK\_TOTAL\_PARTS**) + moneyCashback;  
 moneySpentTotal += moneyCashback \* **CASHBACK\_TOTAL\_PARTS**;  
 }  
  
 moneySpentTotal += moneyTotal;  
  
 printf("%lld\n", moneySpentTotal);  
 }  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/1296B.c>

17. Проблемные обеды (276A)

#include <stdio.h>  
  
long long max(long long a, long long b) {  
 return a > b ? a : b;  
}  
  
int main() {  
 int restaurantAmount, availableTime;  
 scanf("%d %d", &restaurantAmount, &availableTime);  
  
 int restaurantPleasure, lunchTime;  
 scanf("%d %d", &restaurantPleasure, &lunchTime);  
  
 int maxPleasure = restaurantPleasure - max(lunchTime - availableTime, 0);  
  
 for (int currentRestaurant = 1; currentRestaurant < restaurantAmount;

currentRestaurant++) {  
 scanf("%d %d", &restaurantPleasure, &lunchTime);  
  
 int currentPleasure = restaurantPleasure –

max(lunchTime - availableTime, 0);  
 maxPleasure = max(maxPleasure, currentPleasure);  
 }  
  
 printf("%d", maxPleasure);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/276A.c>

18. Хитрая сумма (598A)

#include <stdio.h>  
  
long long log2(long long a) {  
 int power = 0;  
  
 while (a > 1) {  
 a /= 2;  
 power++;  
 }  
  
 return power;  
}  
  
long long twoInPow(long long pow) {  
 return 1 << pow;  
}  
  
int main() {  
 int taskAmount;  
 scanf("%d", &taskAmount);  
  
 for (int currentTask = 0; currentTask < taskAmount; currentTask++) {  
 long long number;  
 scanf("%lld", &number);  
  
 long long powerOfMaxNumberThatTwoInPower = log2(number);  
 long long sumOfNumbersTwoInPower = twoInPow(powerOfMaxNumberThatTwoInPower +

1) - 1;  
 long long sumOfNumbers = ((1 + number) \* number) / 2;  
 long long sumOfNumbersSubtractSumOfTwoInPower = sumOfNumbers - 2 \*

sumOfNumbersTwoInPower;  
  
 printf("%lld\n", sumOfNumbersSubtractSumOfTwoInPower);  
 }  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/598A.c>

19. Арья и Бран (839A)

#include <stdio.h>  
  
#define **MAX\_CANDIES\_TO\_GIVE\_PER\_DAY** 8  
  
long long min(long long a, long long b) {  
 return a > b ? b : a;  
}  
  
int main() {  
 int daysToGiveCandies, requiredCandiesToGive;  
 scanf("%d %d", &daysToGiveCandies, &requiredCandiesToGive);  
  
 int candiesStored = 0;  
 int currentDay = 0;  
  
 while (currentDay < daysToGiveCandies && requiredCandiesToGive > 0) {  
 int addedCandies;  
 scanf("%d", &addedCandies);  
  
 candiesStored += addedCandies;  
 int givenCandies = min(candiesStored, **MAX\_CANDIES\_TO\_GIVE\_PER\_DAY**);  
 candiesStored -= givenCandies;  
 requiredCandiesToGive -= givenCandies;  
 currentDay++;  
 }  
  
 if (requiredCandiesToGive > 0)  
 printf("-1");  
 else  
 printf("%d", currentDay);  
  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/839A.c>

20. Денежная система Геральдиона (560A)

#include <stdio.h>  
#include <stdbool.h>  
  
int main() {  
 int valueAmount;  
 scanf("%d", &valueAmount);  
  
 **bool** isOnePresented = **false**;  
  
 for (int currentValue = 0; currentValue < valueAmount && !isOnePresented;

currentValue++) {  
 int value;  
 scanf("%d", &value);  
  
 isOnePresented = value == 1;  
 }  
  
 if (isOnePresented)  
 printf("-1");  
 else  
 printf("1");  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/560A.c>

21. Медведь и малина (385A)

#include <stdio.h>  
  
long long max(long long a, long long b) {  
 return a > b ? a : b;  
}  
  
int main() {  
 int daysAmount, raspberryKilosForBorrowing;  
 scanf("%d %d", &daysAmount, &raspberryKilosForBorrowing);  
  
 int previousDayPrice, currentDayPrice;  
 scanf("%d %d", &previousDayPrice, &currentDayPrice);  
  
 int maxProfit = max(0, previousDayPrice - currentDayPrice –

raspberryKilosForBorrowing);  
  
 for (int currentDay = 2; currentDay < daysAmount; currentDay++) {  
 previousDayPrice = currentDayPrice;  
  
 scanf("%d", &currentDayPrice);  
  
 int currentDayProfit = previousDayPrice - currentDayPrice –

raspberryKilosForBorrowing;  
  
 if (currentDayProfit > maxProfit)  
 maxProfit = currentDayProfit;  
 }  
  
 printf("%d", maxProfit);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/385A.c>

22. Возрастающая последовательность (11A)

#include <stdio.h>  
  
int main() {  
 int numberAmount, addNumber;  
 scanf("%d %d", &numberAmount, &addNumber);  
  
 int number;  
 scanf("%d", &number);  
  
 int previousNumber = number;  
 int addTimesTotal = 0;  
  
 for (int currentNumber = 1; currentNumber < numberAmount; currentNumber++) {  
 scanf("%d", &number);  
  
 int deltaNumbers = previousNumber - number;  
  
 if (deltaNumbers >= 0) {  
 int addTimes = deltaNumbers / addNumber + 1;  
 number += addTimes \* addNumber;  
 addTimesTotal += addTimes;  
 }  
  
 previousNumber = number;  
 }  
  
 printf("%d", addTimesTotal);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/11A.c>

23. Расписание Алёны (586A)

#include <stdio.h>  
  
#define **NOT\_INITED** (-1)  
#define **BREAKS\_ENOUGH\_TO\_LEAVE** 1  
#define **TOO\_MANY\_BREAKS** 2  
  
int main() {  
 int classAmount;  
 scanf("%d", &classAmount);  
  
 int breakCounter = **TOO\_MANY\_BREAKS**;  
 int totalVisitedClasses = 0;  
  
 for (int currentClass = 0; currentClass < classAmount; currentClass++) {  
 int isClassPresented;  
 scanf("%d", &isClassPresented);  
  
 if (isClassPresented) {  
 totalVisitedClasses++;  
 breakCounter = 0;  
 } else if (breakCounter < **BREAKS\_ENOUGH\_TO\_LEAVE**) {  
 totalVisitedClasses++;  
 breakCounter++;  
 } else if (breakCounter == **BREAKS\_ENOUGH\_TO\_LEAVE**) {  
 totalVisitedClasses--;  
 breakCounter = **TOO\_MANY\_BREAKS**;  
 }  
 }  
  
 totalVisitedClasses -= (breakCounter != **TOO\_MANY\_BREAKS**) \* breakCounter;  
  
 printf("%d", totalVisitedClasses);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/586A.c>

24. \*\* Нечётная сумма (797B)

Было найдено два решения задачи: достаточное для конкретной задачи и более универсальное, с использованием динамического программирования. Оба варианта решения оказались верными.

#include <stdio.h>  
  
#define **NOT\_INITED** 0  
  
long long max(long long a, long long b) {  
 return a > b ? a : b;  
}  
  
int main() {  
 int numbersAmount;  
 scanf("%d", &numbersAmount);  
  
 int oddSum = 0;  
 int minPositiveOdd = **NOT\_INITED**;  
 int maxNegativeOdd = **NOT\_INITED**;  
  
 for (int currentNumber = 0; currentNumber < numbersAmount; currentNumber++) {  
 int number;  
 scanf("%d", &number);  
  
 if (number > 0)  
 oddSum += number;  
  
 if (number % 2 != 0) {  
 if (number > 0 && (number < minPositiveOdd ||

minPositiveOdd == **NOT\_INITED**))  
 minPositiveOdd = number;  
 else if (number < 0 && (number > maxNegativeOdd ||

maxNegativeOdd == **NOT\_INITED**))  
 maxNegativeOdd = number;  
 }  
 }  
  
 if (oddSum % 2 == 0) {  
 if (minPositiveOdd != **NOT\_INITED** && maxNegativeOdd != **NOT\_INITED**)  
 oddSum = max(oddSum - minPositiveOdd, oddSum + maxNegativeOdd);  
 else if (minPositiveOdd != **NOT\_INITED**)  
 oddSum -= minPositiveOdd;  
 else  
 oddSum += maxNegativeOdd;  
 }  
  
 printf("%d", oddSum);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/797B_1.c>****

#include <stdio.h>  
  
#define **NOT\_INITED** 0  
  
int main() {  
 int numbersAmount;  
 scanf("%d", &numbersAmount);  
  
 int oddSum = **NOT\_INITED**;  
 int evenSum = 0;  
  
 for (int currentNumber = 0; currentNumber < numbersAmount; currentNumber++) {  
 int number;  
 scanf("%d", &number);  
  
 int oldEvenSum = evenSum;  
  
 if (oddSum != **NOT\_INITED**) {  
 int addedSum = oddSum + number;  
  
 if (addedSum % 2 && addedSum > oddSum) {  
 oddSum = addedSum;  
 } else if (addedSum % 2 == 0 && addedSum > evenSum) {  
 evenSum = addedSum;  
 }  
 }  
  
 int addedSum = oldEvenSum + number;  
  
 if (addedSum % 2 && (addedSum > oddSum || oddSum == **NOT\_INITED**)) {  
 oddSum = addedSum;  
 } else if (addedSum % 2 == 0 && addedSum > evenSum) {  
 evenSum = addedSum;  
 }  
 }  
  
 printf("%d", oddSum);  
  
 return 0;  
}

Ссылка на репозиторий: <https://github.com/IAmProgrammist/programming-and-algorithmization-basics/blob/c/lab5/797B_2.c>

****

**Код-ревью:**

**Код-ревью на Лабораторную работу №5**

**по теме: «Циклы»**

Работу выполнил: Студента Пахомова Владислава

Провёл ревью: Студент Голуцкий Георгий

**\*Если задача не упомянута, то претензий к ней нет.**

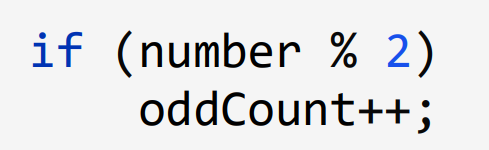
**Задание 2**

* В блок схеме использованы символы % и / вместо mod и div.
* В описании пред. процесса именование самого процесса и переменной в ней совпадают.

**Задание 7**

* Преждевременный ввод переменной contestResult.
* Не смешная шутка :D.

**Задание 8**

*  Можно сократить.

**Задание 10**

* Возможно, стоит решить без применения переменной lastPrimeNumber.

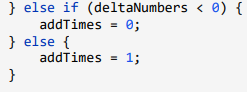
**Задание 15**

* Избыточный расчёт через общее количество пакетов.

**Задание 21**

* Излишняя функция max3. Решение в целом слегка излишне сложное.

**Задание 22**

* Преждевременный ввод переменной number.
* Повторный расчёт переменной deltaNumber.
*  выполняется даже если заведомо ясно что добавлять ничего не надо.
* При ветвлении  else и else if излишне.

**Задание 23**

* Вложенности можно избежать без усложнения условий.
*  Логичнее написать: breakCounter = !isClassPresented \* (breakCouner +1)

**Работа над ошибками (код-ревью):**

**Задание 2**

* Блок схему поправил

**Задание 7**

* Переменную перенёс
* Умею шутить только так

**Задание 8**

* Сократил

**Задание 10**

* Переменную lastPrimeNumber убрал

**Задание 15**

* Исправил, теперь считаю не общее кол-во пакетов, а только неуспешно отправленные пакеты

**Задание 21**

* Упростил код

**Задание 22**

* Упростил условие

**Задание 23**

* Упростил решение задачи

**Вывод:** в ходе выполнения лабораторной работы закреплены навыки написания циклических алгоритмов.