

Title: Hephaestus

Student Names: Alexandru Radac, Codrin Muntean, Daniel Marpozan

School: National College “Octavian Goga”

City and Country: Sibiu, Romania

Category: Robotics

The project description paper

Introduction

The three of us are a Romanian robotics team, named herotech, created in 2018. After the school had begun we decided to participate to GENIUS Olympiad with two projects, one in the Innovation category and this one in the Robotics category.

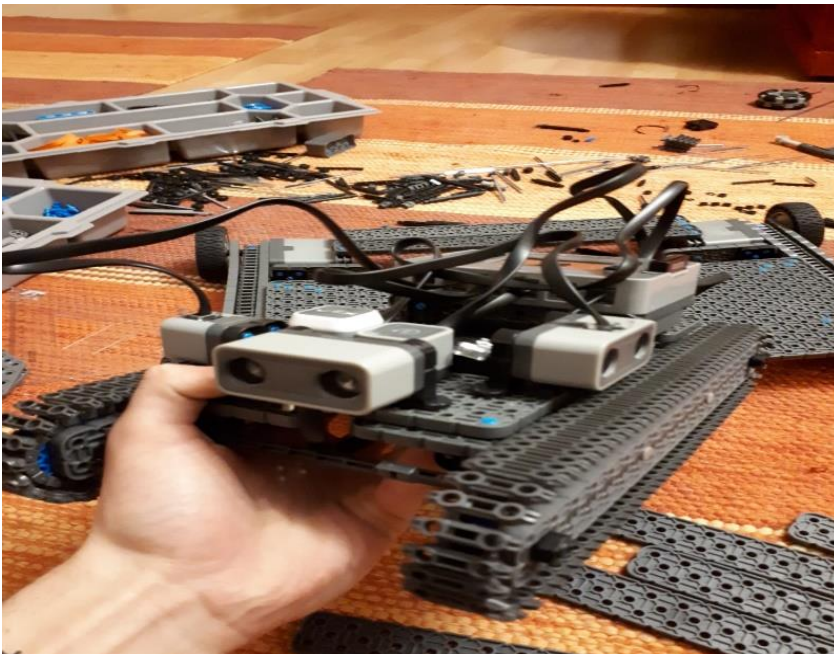


The 3 of us with the finished robot

How it all happened

Our team has created a robot made from VEX IQ components, with a total pieces cost of 900\$. As we didn't have financial support from our school we had to find sponsors ourselves. After about one month of sending e-mails we finally received a sponsorship of 1050\$ from Sifee Action SRL. With that money we payed for the pieces and the transport from USA to Romania.

After we received the components, we started learning how to use them by checking the official designs presented by VEX. To gather even more experience we partially built the robot for the 2018 GENIUS Olympiad. It was a very good idea because after that we understood much better what we had to do and how to do it. We also built a simple robot with all the sensors we had so we could test them and think of how to use them in the future. We started by using the Gyro sensor to make perfect 90 degree turns and by measuring different distances and the errors of the ultrasound distance sensors.



This was our first try: sensors integrated into a simple robot model with tracks.

Once we got a grip of the connections between pieces and motors, and knew what we could do, we started thinking about how to make the robot. We decided that we will build the robot in modules, 8 of them more exactly.

Alex has designed the modules and the algorithm of the first task, Daniel has built the robot and Codrin programmed it. We all worked together as a team and managed to finish the project in 1 week. The final size of the robot is 34cm x 49cm x 17cm.

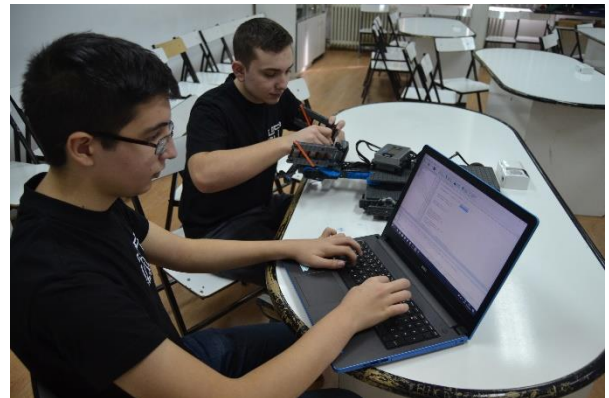


The final form of the robot.

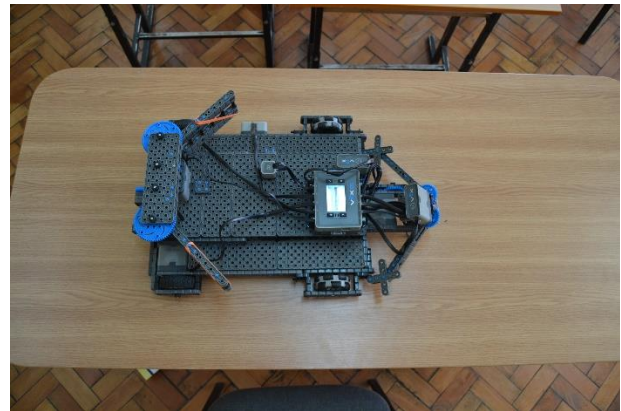
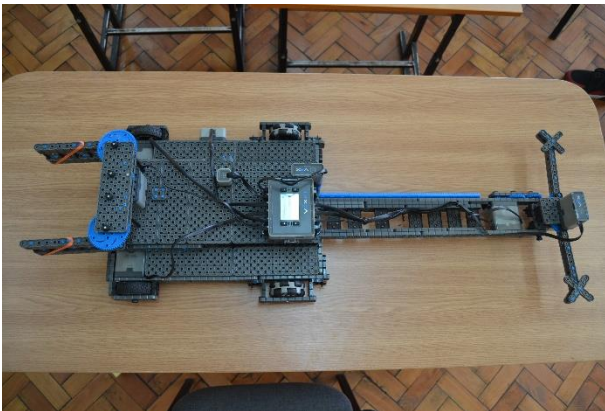
The robot has two claws, one for the foam cubes, and one for the wood figurines. The big claw is fixed onto the robot base, and the small one is fixed onto an extensible arm which we will use in the second task, to pass over the hazard zone.

Everything is connected to the VEX IQ Robot Brain and each one of the big parts can easily be disassembled from the robot.

We usually worked at school and at Alex's place where we created the competition floor design map and where we filmed and tested the robot. We would meet up each day after school at about 5pm and work until 10-11pm.



Alex, Daniel and Codrin finishing the robot in the physics lab.



The robot fully extended and fully contracted

When the robot arm is completely extended, and the claws are positioned at a 90 degrees angle, the robot reaches a total length of 110cm.

We recorded a video showing the robot perform in the first and second task.

Engineering book

Modules

As we said, the robot is made from 8 different modules, that connected make the robot function.

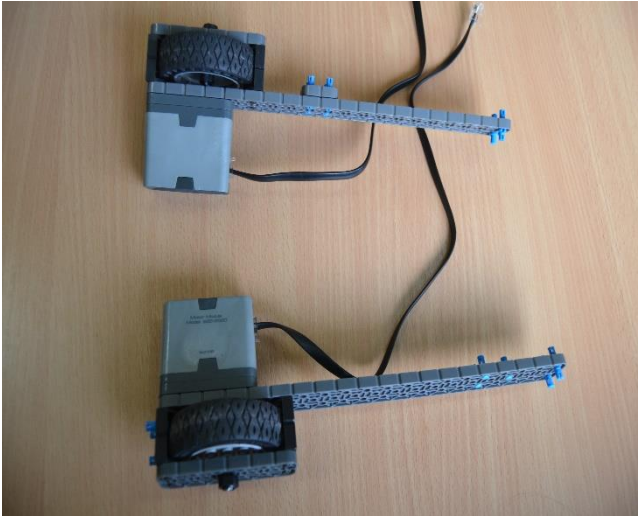


The 8 modules placed side by side

The modules were designed by Alex to be able to complete the necessary tasks as fast as possible, and to be reliable. They were built by Daniel according to the schematics.

The designing/ building phase was the longest, taking about 4 days of the total 7.

Module #1: Motor wheels:

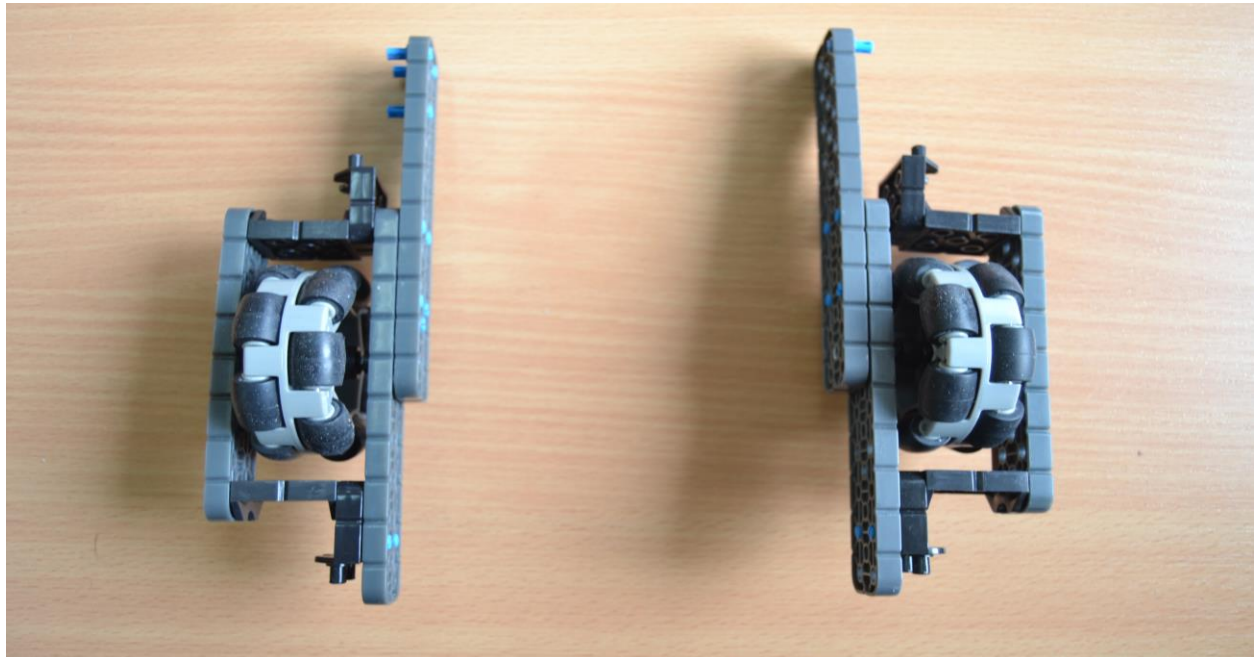


This module is made from 2 wheels directly connected to motors, a support frame that partially envelops the wheels, and 2 longer VEX pieces that are meant to connect the module to the rest of the robot.

At some point we thought to make the robot faster by changing the way the wheels were connected to the motor, more exactly by adding 2 different size gears on each side. The bigger one was connected to the motor and the smaller one was connected to the wheel. Both gears were interconnected, so the robot was moving 3 times faster, as the ration between the gears was 3 to 1.

Unfortunately the robot couldn't turn well if loaded because it didn't have enough force. We will try to make it work in the future months before the actual contest.

Module #2: Omnidirectional wheels:

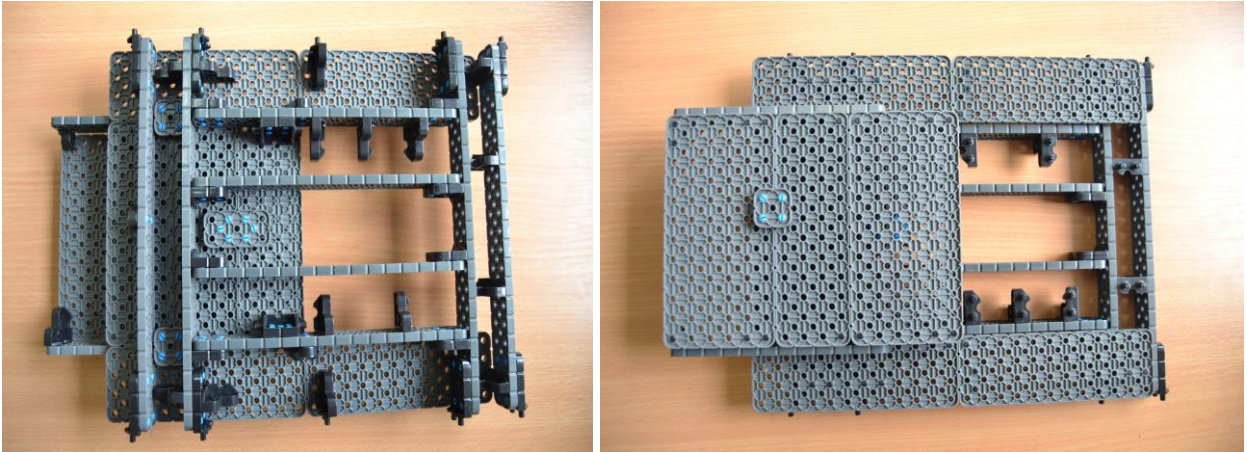


This module is formed from 2 omnidirectional wheels, also partially enveloped in a structure meant to make them more stable, and 2 pieces that connect it to the robot.

We chose this option because our robot is quite long so it was hard for it to make fast turns as the rubber wheels were creating too much friction with the floor.

But there was another problem, because the omnidirectional create nearly 0 friction with the floor, while turning fast, the robot would slide more than the degrees we wanted to. We remediated this problem within the code of the robot, by drastically lowering the speed of the rotations in the last 20 degrees of the turn so the robot would lose its momentum.

Module #3: Base/resistance structure of the robot:



The base of the robot as seen from under and above without any other modules on it.

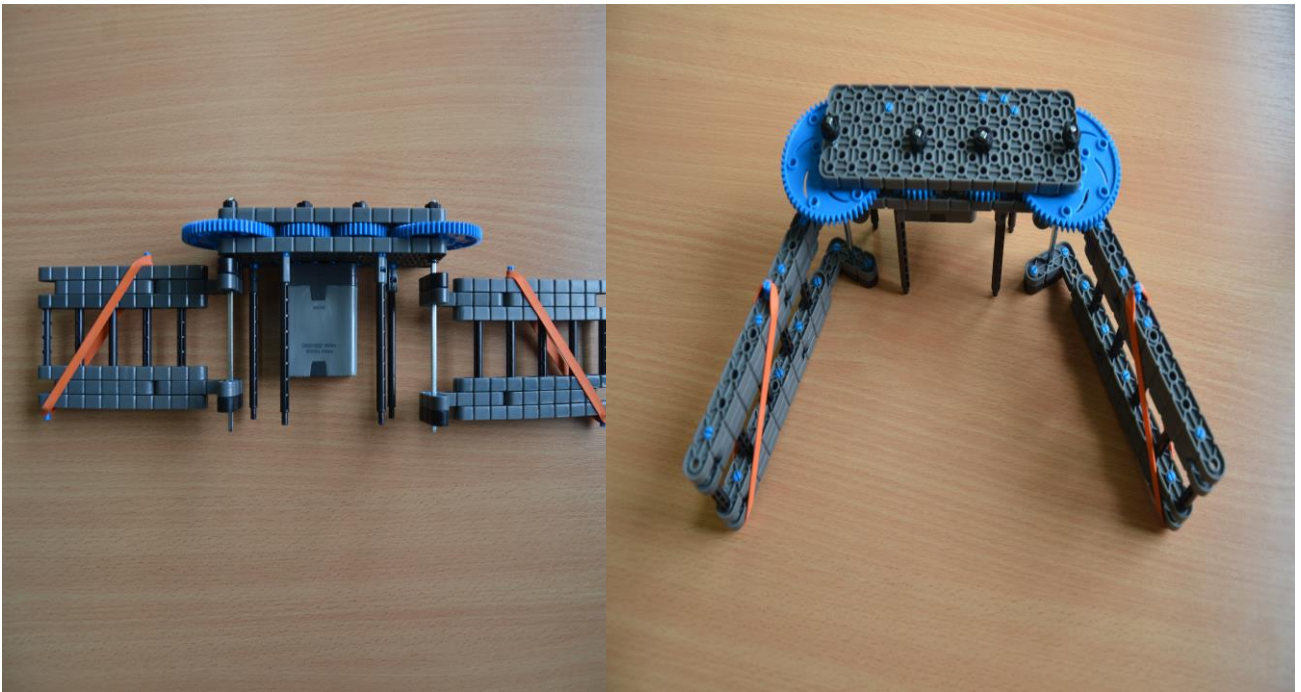
This module is the biggest of all and is the one that supports all of the other. It's made from multiple flat and long VEX components placed in vertical and horizontal positions.

As it can be seen in the first picture, some beams are placed in a 90 degrees angle. That is a decision we took because the robot would “wobble” if we were taking sharp turns. The same reason is why the base is covered in flat pieces.

The open spot is the place where the module #5 is placed. It has a lot of connectors because it has to be very well fixed and perfectly stable.

This is also the only “solid” part of the robot, that has no moving parts and under it there are some cables that connect different motors to the Robot Brain without being in the way of the robot arms.

Module #4: Big claw:



The claw as seen from the front side and above

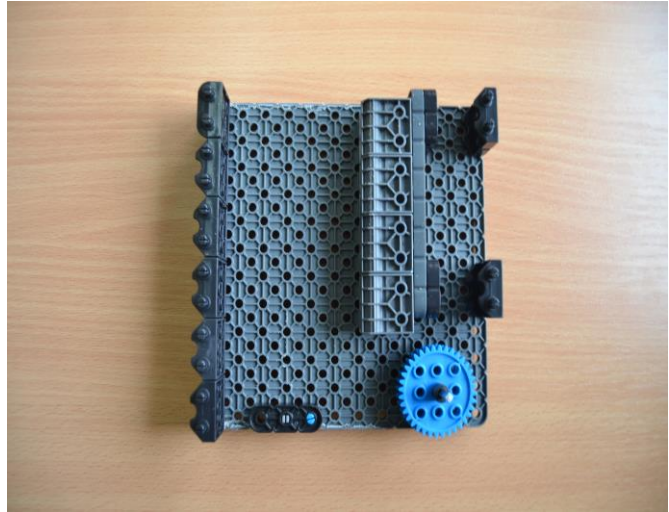
This module is composed of 1 motor, 2 medium size gears and 2 big sized gears, the actual arms that hold the foam cube with a rubber strip and a fixing support.

The motor is connected to one of the medium gears, which passes the circular motion to the big gear at one side and the interconnected medium-big gear at the other side.

The arms are connected to the big gears directly. Through the medium to small gear mechanism, the pressing force of the arms is doubled as the ratio between the gears is 2 to 1.

This was the first module that we designed and we decided to stick with it from the beginning to the end.

Module #5: Long arm housing:



The module as seen from a side and from above without flat cover and motor

This module is quite a simple one. It has 2 big flat components, a guidance piece for the arm, a stopper so the arm can't extend more than we want and 2 medium gears connected to a motor, stacked one on top of each other to move the arm.

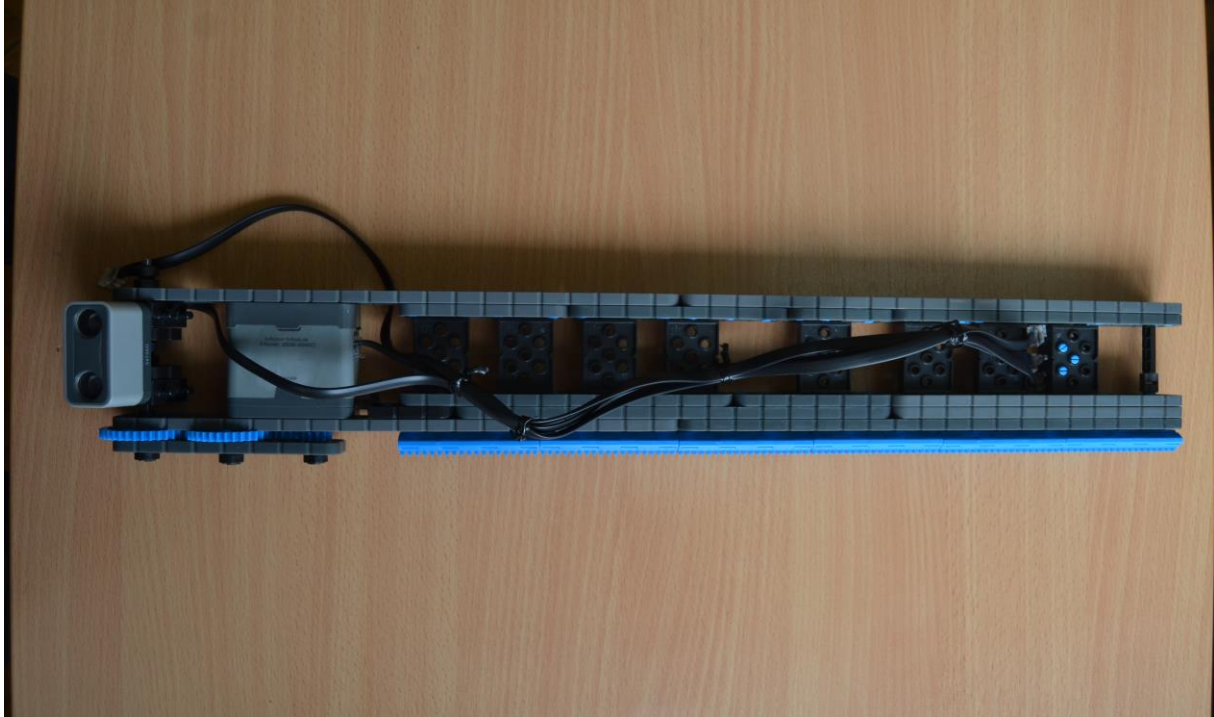
This was also a module we used from the beginning without much modification.

It is very important for it to be solid and to be well fixed to the robot base, because when the long arm (module #6) is extended it must keep it as straight as possible, without letting it overextend and tilt.

We had quite a lot of trouble building this one, because the arm would get stuck in the edges of the big flat parts, so we had to polish them to be smooth.

One problem we encountered was that, when the arm was fully extended, the engine wouldn't have enough power to pull it back in, so we had to run it at maximum speed.

Module #6: The long arm:



The long arm is the most important module of the robot as it is used both in the first task and in the second one. It is composed of a long VEX structure made from beams, connected between them with the black pieces, an engine connected to 4 gears (2 medium, 2 small) that pass the movement and accentuate the force 3 times, and 6 blue “rails” that connect this module to the previous one, and make extension possible. It has a distance sensor that we use in the first task connected to a shaft that is rotated by the last gear and a support for the #7 module.

This module was the hardest one to build as we had a lot of restrictions for weight, length, resistance and the way the cables were placed. We spent a full day building it but we did a good job as we didn't have to modify it later.

One problem we had was that the motor didn't have enough force so we had to add the two small gears to triple the force but that also decreased the rotation speed.

Module #7: The small claw:



The claw as seen from above and behind

This claw went through quite a lot of modifying as we had to increase the maximum distance we could pick up a wood figurine. We decided that the claw should have 2 long arms. We also put at the end of the arms a + form piece was easy to fix and could hold the figurine properly “by the head”.

The module is made from 2 same size medium gears connected to the arms, an engine connected to one of the gears, respectively to the other, as the gears are interconnected.

This module has been designed to be placed at the end of the long arm (module #6).

Initially this module had 2 small gears, but we changed it to have 2 medium gears because that way the figurine was easier to pick and it wouldn't fall at times from the grip.

We will modify this module in the future months so it can have a bigger error margin in the first task, because now, the robot doesn't always pick up the figurine right.

Module #8: Robot Brain and sensors:



This part is not actually a module made by us, but one “customized” by us. It contains the main Robot Brain, a gyroscope sensor and a distance sensor. This module also contains all the 9 cables of different lengths that connect everything to the Robot Brain.

It is placed on top of the robot base, connected in 4 points and has a space to remove the battery without having to take everything out, for a faster and easier charging.

We decided to add a gyroscope to guide the robot to take perfect x degrees turns. We also have a distance sensor that is placed on the side of the robot and is used in the first task.

Programming



We used the RobotC IDE provided on the VEX IQ website. Because we already had a little bit of experience in the programming domain, we opted to use the C++ language instead of the visual language.

First task:

In this task the robot starts oriented with the extensible arm towards the fire-zone. We programmed the robot to move 50cm forwards, to take a 90 degree turn to the right, then go forward 1 meter. While it is moving forward the robot measures the distance with the sensor placed on it's left side. If the robot detects something closer than 40 cm, it means that that was the figurine. We remember the value (the minimum distance detected between the robot and the figurine, that is between 10 and 40 cm) the sensor gave us as the x variable. After this, the

robot takes a 90 degree turn to the left, moves forward 20 centimeters + the x value. This makes the robot to be on the same line as the figurine is. The robot takes another 90 degrees turn to the left and then proceeds to go forward until the sensor placed on the long arm detects something at 14 cm (this means the figurine is at 14cm before the sensor, the same distance the claw closes, so it can catch the wood figurine). We measure how much the robot moves with the help of the encoder included in the motor. We will call the distance variable y. After the robot detects the figurine at 14 cm, it stops, then closes and lifts the claw. Next, the robot moves backwards from the point it started moving to pick up the wood figurine (goes backwards the y value and 60 more cm to be in line with the hospital). After this, the robot takes a 90 degree turn to the left and moves ahead $70 + x$ cm. The robot opens the claw, letting the figurine fall into the bin and then returns to the lot.

The time we got to complete this task was 32 seconds, including the initial 3 seconds in which the gyroscope calibrates. The turns are also limited to a fifth of the maximum speed.

Second task:

We basically just programmed the robot to move at the controller action. We have 2 modes that we can change by pressing the Right-Down button on the controller. We needed 2 modes because we didn't have enough buttons on the controller to do everything we wanted. For the first mode, we consider the front of the robot where the big-claw module is. Using the A-B joystick we can make the robot move and turn. Using the C-D joystick we control the big claw. In the second mode, we reversed all the movement controls, and we can control the extensible arm, the lifting and the clutching of the claw.

Functions:

```
14
15     int wheelDiameterCm = 6.4;
16
17     void moveCm(float distance)
18     {
19
20         float circumference = PI * wheelDiameterCm;
21         float degreesToRotate = (distance * 360) / circumference;
22
23         resetMotorEncoder(leftMotor);
24         resetMotorEncoder(rightMotor);
25
26         moveMotorTarget(leftMotor, degreesToRotate, 100);
27         moveMotorTarget(rightMotor, degreesToRotate, 100);
28
29         waitUntilMotorStop(leftMotor);
30         waitUntilMotorStop(rightMotor);
31
32         wait1Msec(100);
33     }
34
```

This function transforms an input distance from centimeters into rotation degrees, that the motor can use. The formula is $(\text{distance_variable} \times 360) / (\text{PI} \times 6.4)$.

```
120
121     void rotate(int degrees)
122     {
123         int currentRotationDegrees = getGyroDegrees(gyroSensor);
124
125         int direction = 1;
126         if(degrees < 0) direction = -1;
127
128         while(getGyroDegrees(gyroSensor) < degrees)
129         {
130             displayTextLine(3, "Gyro Value is: %d", getGyroHeading(gyroSensor));
131
132             if(getGyroDegrees(gyroSensor) < currentRotationDegrees - 25)
133             {
134                 setMotorSpeed(leftMotor, (-75 * direction));
135                 setMotorSpeed(rightMotor, (75 * direction));
136             }
137             else
138             {
139                 setMotorSpeed(leftMotor, (-20 * direction));
140                 setMotorSpeed(rightMotor, (20 * direction));
141             }
142         }
143     }
144 }
145
```

This function makes the turns go slower when they reach the final 25 degrees, so the robot will loose its angular momentum and will not slide when the rotation is complete.