Додаток Б

(обов'язковий)

Текст програми transmission.py

```
def int_to_gf(n, F, bin_length):
    F2 = GF(2)
    def chr_to_gf2(c):
        if c == '0':
            return F2.zero()
        else:
            return F2.one()
    bin_str = bin(n)[2:].rjust(bin_length, '0')
    if F.base_ring() != F2:
        raise Exception('Field must be an extention of GF(2)')
    bits_per_gf_element = F.vector_space().dimension()
    if bin_length % bits_per_gf_element != 0:
        raise Exception('Binary length is not adjusted to F size')
    number_of_gf_elements = bin_length / bits_per_gf_element
    result = []
    for i in range(number_of_gf_elements):
        result.append(
                F(vector(F2, [chr_to_gf2(c) for c in
                    bin_str[i*bits_per_gf_element:(i+1)*bits_per_gf_element]]
                    ))
                    )
    return result

def gf_to_int(gf_list):
    bin_list = []
    for element in gf_list:
        bin_list += [str(c) for c in element._vector_()]
    return int(''.join(bin_list), 2)

def server(HC, PORT=50007, text='Hello world', p_err=0.01):
    # HC_a = HermitianCode(4, 53)
    # HC = HermitianCode(4, 37)
    # server(HC_a,text=open('/tmp/lorem.txt').read(), PORT=10001, p_err=0.3)
    import socket
    HOST = ''
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.settimeout(5)
    s.bind((HOST, PORT))
    s.listen(1)
    conn, addr = s.accept()
    print 'Connected by', addr
    i = 0
    while i < len(text):
        data = conn.recv(1024)
        if not data:
            break
        message = []
        while len(message) != HC.k:
            try:
                message += int_to_gf(ord(text[i]), HC.F, 8)
            except IndexError:
                message += int_to_gf(ord('~'), HC.F, 8)
            i += 1
        codeword = HC.encode(message)
        for j in range(len(codeword)):
            if random() < p_err:
                codeword[j] += HC.F.random_element()
        conn.send(str(gf_to_int(codeword)))
    conn.close()


def client(HC, PORT=50007):
    # client(HC_a, PORT=10001)
    import socket, sys

    HOST = 'localhost'
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.settimeout(5)
    s.connect((HOST, PORT))
    while 1:
        s.send('Ok')
        data = s.recv(1024)
        if not data:
```

```
        break
    else:
        received_word = int_to_gf(
                int(data),
                HC.F,
                HC.F.vector_space().dimension()*HC.n)
        #print received_word
        try:
            decoded_word = list(HC.decode(vector(HC.F, received_word)))
            n_gf_per_c = 8/HC.F.vector_space().dimension()
            for i in range(len(decoded_word)/n_gf_per_c):
                sys.stdout.write(chr(gf_to_int(
                    decoded_word[i*n_gf_per_c:(i+1)*n_gf_per_c]
                    )))
                sys.stdout.flush()
        except DecodingError:
            for i in range(HC.k*HC.F.vector_space().dimension()/8):
                sys.stdout.write('_')
            sys.stdout.flush()
print
s.close()
```